



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **10134175 A**

(43) Date of publication of application: **22.05.98**

(51) Int. Cl.

G06T 3/40

G06F 3/14

(21) Application number: **08286571**

(22) Date of filing: 29.10.96

(71) Applicant: **SONY CORP**

(72) Inventor: IWASE SEIICHIRO
KUROKAWA MASUYOSHI
KANO MAMORU
NAKAMURA KENICHIRO

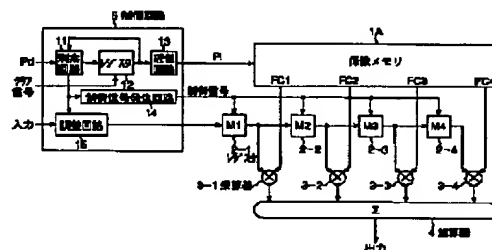
(54) PROCESSOR AND METHOD FOR IMAGE PROCESSING

(57) Abstract:

PROBLEM TO BE SOLVED: To enlarge or reduce an image at an arbitrary conversion rate.

SOLUTION: A residue circuit 11 outputs a phase variation component Pd supplied from a specific device and the decimal part of the sum of the value of a register 12 to the register 12. An approximation circuit 13 outputs a filter select signal Pi whose phase (x) corresponds to a filter coefficient set corresponding to the phase closest to the value of the register 12 to a coefficient memory 1A. Thus, an optimum filter coefficient set among a specific number of filter coefficient sets is selected for the interpolation of specific pixel data. Then product sum operations between the four filter coefficient sets and four pixel data are performed by multipliers 3-1 to 3-4 and an adder by a Cubic approximating method to calculate an interpolated value of pixels.

COPYRIGHT: (C)1998,JPO



THIS PAGE BLANK (USPTO)

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-134175

(43) 公開日 平成10年(1998) 5月22日

(51) Int.Cl.⁶

識別記号

F I

G 0 6 T 3/40

G 0 6 F 15/66

3 5 5 C

G 0 6 F 3/14

3 1 0

3/14

3 1 0 A

審査請求 未請求 請求項の数19 O L (全 31 頁)

(21) 出願番号

特願平8-286571

(22) 出願日

平成8年(1996)10月29日

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 岩瀬 清一郎

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 黒川 益義

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(72) 発明者 加納 健

東京都品川区北品川6丁目7番35号 ソニー株式会社内

(74) 代理人 弁理士 稲本 義雄

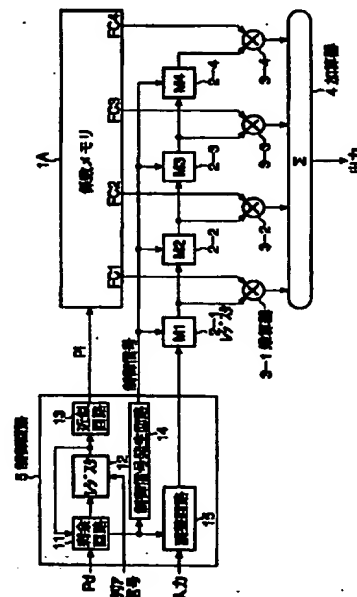
最終頁に続く

(54) 【発明の名称】 画像処理装置および方法

(57) 【要約】

【課題】 任意の変換比率の画像の拡大または縮小を行うことができるようにする。

【解決手段】 剰余回路11は、所定の装置より供給された位相変化分P dと、レジスタ12の値の和の小数部分をレジスタ12に出力する。そして、近似回路13は、位相xが、レジスタ12の値に最も近い位相に対応するフィルタ係数セットに対応するフィルタ選択信号P iを係数メモリ1Aに出力する。このようにして、所定の画素データの補間において、所定の数のフィルタ係数セットのうちの最適なフィルタ係数セットが選択される。そして、Cubic近似法に従って、その4つのフィルタ係数セットと、4つの画素データとの積和演算が、乗算器3-1乃至3-4と加算器4で行われることにより、画素の補間値が算出される。



【特許請求の範囲】

【請求項1】 画像の拡大または縮小に伴う画素データの補間演算を、その補間演算に対応したハードウェアで行う演算部と、前記補間演算に利用されるフィルタ係数セットを記憶する記憶部とを備える画像処理装置で、前記記憶部は、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを記憶し、そのフィルタ係数のうち、前記位相が、前記補間演算される画素データの位相に最も近いフィルタ係数セットを前記演算部に出力し、前記演算部は、前記フィルタ係数セットを利用して、前記画素データの補間演算を行うことを特徴とする画像処理装置。

【請求項2】 前記分割数は、2のべき乗であることを特徴とする請求項1に記載の画像処理装置。

【請求項3】 画像の拡大または縮小に伴う画素データの補間演算を、その補間演算に対応したハードウェアで行う演算部と、前記補間演算に利用されるフィルタ係数セットを記憶する記憶部とを備える画像処理装置における画像処理方法で、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを記憶している前記記憶部から、前記位相が、前記補間演算される画素データの位相に最も近いフィルタ係数セットを前記演算部に出力し、前記演算部により、前記フィルタ係数セットを利用して、前記画素データの補間演算を行うことを特徴とする画像処理方法。

【請求項4】 複数の要素プロセッサに画素データを供給し、前記複数の要素プロセッサをSIMD制御して、各要素プロセッサで、周辺の要素プロセッサが有する画素データを利用して、画像の拡大または縮小に伴う画素データの補間の処理を並列に行う画像処理装置で、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、前記位相が、処理される画素データの位相に最も近いフィルタ係数セットを、前記要素プロセッサにそれぞれ供給し、前記要素プロセッサが、前記フィルタ係数セットを利用して、前記画素データの補間の処理をそれぞれ行うことを特徴とする画像処理装置。

【請求項5】 前記分割数は、2のべき乗であることを特徴とする請求項4に記載の画像処理装置。

【請求項6】 前記要素プロセッサは、データを1ビットずつ処理する1ビットプロセッサであることを特徴とする請求項4に記載の画像処理装置。

【請求項7】 前記周辺の要素プロセッサと、前記所定の要素プロセッサとの位置関係のパターンの種類が最小になるように、前記複数の要素プロセッサに前記画素データを供給することを特徴とする請求項4に記載の画像処理装置。

【請求項8】 前記フィルタ係数セットは、前記画素データが供給されるときに利用される回路を介して、前記要素プロセッサに供給されることを特徴とする請求項4に記載の画像処理装置。

【請求項9】 前記要素プロセッサに接続され、前記フィルタ係数セットを記憶する記憶手段をさらに備えることを特徴とする請求項8に記載の画像処理装置。

【請求項10】 前記記憶手段は、前記フィルタ係数セットを、前記フィルタ係数セットに対応する位相の順番に従って記憶することを特徴とする請求項9に記載の画像処理装置。

【請求項11】 前記要素プロセッサは、前記フィルタ係数セットを記憶する記憶部と、演算を行うALU部を備え、

各要素プロセッサに割り当てられる画素データの位相情報に対応した前記フィルタ係数セットが、前記ALU部を介して前記記憶部に供給されることを特徴とする請求項4に記載の画像処理装置。

【請求項12】 前記要素プロセッサが前記位相情報をそれぞれ算出することを特徴とする請求項11に記載の画像処理装置。

【請求項13】 前記ALU部に接続され、前記フィルタ係数セットを記憶する記憶手段をさらに備え、前記記憶手段は、前記フィルタ係数セットを、前記フィルタ係数セットに対応する位相の順番に従って記憶することを特徴とする請求項11に記載の画像処理装置。

【請求項14】 前記要素プロセッサは、その要素プロセッサに割り当てられる画素データの位相情報に対応して、補間に利用されるフィルタ係数セットをそれぞれ算出することを特徴とする請求項4に記載の画像処理装置。

【請求項15】 前記要素プロセッサが前記位相情報をそれぞれ算出することを特徴とする請求項13に記載の画像処理装置。

【請求項16】 前記補間演算は、Cubic近似に対応する演算であることを特徴とする請求項4に記載の画像処理装置。

【請求項17】 前記補間演算として、第1の補間方式に従って第1の分割数で第1の補間演算を行った後、前記第1の補間演算の演算結果に対して、第2の補間方式に従って第2の分割数で第2の補間演算を行うことを特徴とする請求項4に記載の画像処理装置。

【請求項18】 前記画素データは、輝度データと色データで構成され、

前記色データは、前記輝度データの補間を行うときに利用されるフィルタ係数セットに対応する前記分割数より少ない分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを利用して補間されることを特徴とする請求項4に記載の画像処理装置。

【請求項19】 複数の要素プロセッサに画素データを

供給し、前記複数の要素プロセッサをSIMD制御して、各要素プロセッサで、周辺の要素プロセッサが有する画素データを利用して、画像の拡大または縮小に伴う画素データの補間の処理を並列に行う画像処理方法で、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、前記位相が、処理される画素データの位相に最も近いフィルタ係数セットを、前記要素プロセッサにそれぞれ供給し、前記要素プロセッサが、前記フィルタ係数セットを利用して、前記画素データの補間の処理をそれぞれ行うことを特徴とする画像処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、画像処理装置および方法に関し、特に、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、補間演算される画素の位相に最も近いフィルタ係数セットを利用して、画素データの補間演算を行うことにより、任意の倍率の画像の拡大または縮小を行う画像処理装置および方法に関する。

【0002】

【従来の技術】テレビジョン受像機などの画像表示器（ディスプレイ）は、CRT（CathodeRay Tube）を使用しているものが多い。このようなディスプレイにおいて、各種画像方式に対応してアナログ的に画像信号を取り扱い、画像を表示する場合、水平走査周波数を変えて対応することが多い。

【0003】一方、デジタル的に画像信号を取り扱う場合、NTSCやPALなどの放送伝送方式に応じて、画像の解像度が異なるので、それらの方式でデジタル化された画像の縦方向と横方向の画素数は、放送伝送方式毎に異なる。また、放送伝送方式としては、HDTVまで含めていろいろな方式があり、その画素数（解像度）の規格は多様である。そのため、画像データに対してデジタル処理を行うシステムにおいては、これらのすべての伝送方式に対応させる場合、画素数を「補間フィルタ」で変換する必要がある。

【0004】さらに、最近普及している液晶ディスプレイやプラズマディスプレイにおいては、表示画像の画素数が所定の数に固定されているため、このようなディスプレイを利用する場合においても、原画像の画素数を、これらのディスプレイに対応する画素数に変換するために補間フィルタが必要となる。

【0005】次に、画像の画素数を変換するための補間フィルタの一例について説明する。

【0006】まず、画像の拡大または縮小、および、標本化周波数（画素数）の変換について説明する。

【0007】画像の拡大または縮小と、画像の標本化周波数（画素数）の変換（解像度の異なる画像規格間の変換）は、どちらも、原画像の各画素位置に対して、元の

画像において存在しなかった画素のデータを求める演算を行うことにより実現される。従って、同様の演算操作を行う補間フィルタを利用することにより、上述の2つの処理を行うことが可能である。

【0008】図21は、原画像の一部分の一例を示している。図中の丸印は画素の位置を表している。この部分においては、横方向に8画素、縦方向に6画素が含まれている（便宜上、ここでは画素数を小さい値に設定している）。

10 【0009】次に、この原画像を例えば（10/7）倍に拡大する場合について説明する。なお、倍率は面積ではなく長さの比で表現するものとする。図21の画像を拡大する場合、表示画像規格は変えずに、画素の配列（即ち画素間隔など）を、図21と同一に保つ。そのようにして、拡大の処理を行った場合、その結果の画像は、図22のようになる。この場合、倍率は、1.429（=10/7）であるので、画像の1辺の長さは、1.429倍され、画素数は、約1.429²倍に増加する。

20 【0010】例えば水平方向（水平走査の方向）に対して、原画像においては画素数が8であるが、拡大後には、11または12（8×10/7=11.429に近い整数）画素になる。従って、拡大後の相似画像における画像の同じ部分に対応する各画素の位置関係は、原画像における位置関係とは異なるので、拡大後の各画素のデータ（輝度や色を表現する）の値は、原画像のそれとは異なることになる。

【0011】図23は、（10/7）倍の倍率で、画像の拡大した場合における、原画像と拡大後の画像における水平方向の画素の位置関係について示している。

30 【0012】図中、上側のR_i（i=1, 2, ...）は、原画像の画素を表しており、下側のQ_i（i=1, 2, ...）は、拡大後の補間画素のデータを表している。R_iに対応する画素は、Q_iに対応する画素の間隔の（10/7）倍の間隔で配置されている。なお、図23は、水平方向の拡大の様子だけを示しているが、垂直方向についても同様であるので、その説明は省略する。

【0013】拡大後の各画素のデータの値は、図23に示すような原画像の各画素の位置との対応関係に応じ、周辺のいくつかの原画像の画素の値から、補間フィルタ演算、即ち補間関数の畳み込み演算を行うことにより算出する（後述）。

40 【0014】次に、画像の大きさを変化させずに、標本化周波数を例えば（10/7）倍にする場合を考える。この標本化周波数変換は、解像度が（10/7）倍だけ高い画像規格に変換することと等価である。即ち、水平方向の画素数は、（10/7）倍に変更される。この場合、図21の原画像は、図24に示すように、1次元的には約1.429倍の画素数、即ち1.429²倍の面密度を有する画像に変換される。

【0015】図21の各画素と図22の各画素との対応関係と、図21の各画素と図24の各画素との対応関係は、両者とも、図23に示すようになり、同一であるので、画素数の多い画像規格に変換する演算操作は、上述の画像の拡大の演算操作と同様に行われる。

【0016】次に、図21の原画像を例えば(10/13)倍に縮小する場合について説明する。

【0017】画像の縮小を行う場合、画像の規格は変えないので、縮小後の画像における画素の配列、即ち画素間隔などは、図21に示す原画像と同一になる。

【0018】図25は、図21の原画像を、(10/13)倍に縮小した画像を示している。この場合、倍率は、0.769(=10/13)であるので、画像の1辺の長さは、0.769倍に縮小し、縮小画面を構成する画素数は、約0.769²に減少する。

【0019】例えば、原画像においては水平方向の画素数が8であるが、縮小後の画像においては、水平方向の画素数は、6または7(8×10/13=6.154に近い整数)になる。従って、縮小後の相似画像における画像の同じ部分に対応する各画素の位置関係は、原画像における各画素の位置関係とは異なるので、縮小後の各画素のデータ(輝度や色を表現する)の値は、原画像のそれとは異なる。

【0020】図26は、(10/13)倍の倍率で、画像を縮小した場合における、原画像と縮小後の画像における水平方向の画素の位置関係について示している。

【0021】図中、上側のR_i(i=1, 2, ...)は、原画像の画素を表しており、下側のQ_i(i=1, 2, ...)は、縮小後の補間画素のデータを表している。R_iに対応する画素は、Q_iに対応する画素の間隔の(10/13)倍の間隔で配置されている。なお、図26は、水平方向の縮小の様子だけを示しているが、垂直方向についても同様であるので、その説明は省略する。

【0022】縮小後の各画素のデータの値は、図26に*

$$f(x) = \text{sinc}(\pi \times x) = \sin(\pi \times x) / (\pi \times x) \quad (1)$$

【0029】ここで、 π は、円周率を表す。

【0030】しかしながら、実際には、有限時間内に補間値を算出する必要があるので、sinc関数を、有限の範囲で近似した補間関数を利用する。

【0031】近似の方法としては、最近傍近似法、双一次近似法、Cubic近似法などが知られている。

【0032】最近傍近似法においては、式(2)および図29(B)に示すような補間関数を利用して、原画像の1画素のデータから、補間後の1画素のデータを演算する。なお、式(2)および図29(B)の変数 x は、原画像の画素位置からの水平方向の変位を、原画像の標本間隔で正規化した量を表すものとする。

【数1】

*示すような原画像の各画素との対応関係に応じて、周辺のいくつかの原画像の画素の値から、補間フィルタ演算、即ち補間関数の畳み込み演算を行うことにより算出する。

【0023】次に、画像の大きさを変化させずに、標本化周波数を例えば(10/13)倍にする場合を考える。この標本化周波数変換は、解像度が(10/13)倍だけ低い画像規格に変換することと等価である。即ち、画素数は、(10/13)倍に変更される。この場合、図21の原画像は、図27に示すように、1次元的には約0.769倍の画素数、即ち0.769²倍の面密度を有する画像に変換される。

【0024】図21の各画素と図25の各画素との対応関係と、図21の各画素と図27の各画素との対応関係は、いずれも図26に示すようになり、同一であるので、解像度が低い画像規格に変換する演算操作は、上述の画像の縮小の演算操作と同様に行われる。

【0025】以上のように、画像の拡大または縮小、および、標本化周波数(画素数)の変換を行う場合、原画像には存在しなかった位置の画素データを算出する補間フィルタが必要となる。

【0026】次に、補間フィルタにおいて行われる演算について説明する。

【0027】図28に示すように、原画像の標本化間隔を S とし、原画像の画素 R の位置から距離(位相) P だけ離れた位置を、補間により生成する画素 Q_i の位置(補間点)とすると、画素 Q_i の値は、その周辺の原画像の画素の値 R に対する畳み込み演算により算出される。

【0028】「標本化定理」によれば、理想的な「補間」を行う場合、式(1)および図29(A)に示すようなsinc関数を補間関数 $f(x)$ として、無限時間過去の画素から無限時間将来の画素までの畳み込み演算を行う。

$$\begin{cases} f(x)=1 & -0.5 < x \leq 0.5 \\ f(x)=0 & -0.5 \geq x, x > 0.5 \end{cases}$$

... (2)

【0033】双一次近似法においては、式(3)および図29(C)に示すような補間関数を利用して、原画像の2画素のデータから、補間後の1画素のデータを演算する。なお、式(3)および図29(C)の変数 x は、原画像の画素位置からの水平方向の変位を、原画像の標本間隔で正規化した量を表すものとする。また、双一次近似法は、線形補間として良く知られており、加重平均が算出される。

【数2】

$$\begin{cases} f(x)=1-|x| & |x|\leq 1 \\ f(x)=0 & |x|>1 \end{cases}$$

... (3)

【0034】Cubic近似法においては、式(4)および図29(D)に示すような補間関数を利用して、原画像の4画素のデータから、補間後の1画素のデータを演算する。なお、式(4)および図29(D)の変数xは、原画像の画素位置からの水平方向の変位を、原画像の標本間隔で正規化した量を表すものとする。

【数3】

$$\begin{cases} f(x)=|x|^3-2|x|^2+1 & |x|\leq 1 \\ f(x)=-|x|^3+5|x|^2-8|x|+4 & 1<|x|\leq 2 \\ f(x)=0 & 2<|x| \end{cases}$$

... (4)

【0035】これらの畳み込み演算は、所謂FIRデジタルフィルタを利用して行うことが可能である。その場合、補間関数の中心を補間点に合わせ、所定の画素数分だけ近傍の原画像の標本点で補間関数を標本化した値を補間フィルタ係数セットとして使う。

【0036】例えば、双一次近似法で補間の演算を行う場合、位相Pが0.0であるとき、フィルタ係数セットを構成する2つの重み(フィルタ係数)は、1.0と0.0となり、位置が一致する原画像の画素値をそのまま出力するような係数セットとなる。

【0037】また、位相Pが0.5であるとき、2つのフィルタ係数は、0.5と0.5となり、Pが0.3であるときにおいて、0.7と0.3となる。

【0038】Cubic近似法で補間の演算を行う場合、位相Pが0.0であるとき、フィルタ係数セットを構成する4つの重み(フィルタ係数)は0.0、1.0、0.0、0.0、および、0.0となり、位置が一致する原画像画素のデータ値をそのまま出力するような係数セットとなる。

【0039】また、位相Pが0.5であるとき、4つのフィルタ係数は、-0.125、0.625、0.625、および、-0.125となり、Pが0.3であるときにおいては-0.063、0.847、0.363、および、-0.147となる。

【0040】なお、このとき、データを算出する補間点毎に、原画像の画素との位相Pがそれぞれ異なるので、異なる位相に対応する複数のフィルタ係数のセットが必要となる。

【0041】次に、従来の補間フィルタ演算装置について説明する。

【0042】図30は、補間の演算、即ち補間関数の畳み込み演算を行うFIRデジタルフィルタを利用した演算装置の一構成例を示している。なお、図30の演算装

置は、Cubic近似法を利用して畳み込み演算を行う。

【0043】係数メモリ1は、各補間点(または、各位相)に対応する複数のフィルタ係数を保持しており、所定の装置(図示せず)より供給されるフィルタ選択信号に対応した4つのフィルタ係数FC1、FC2、FC3、FC4を乗算器3-1乃至3-4にそれぞれ出力する。

【0044】レジスタ2-1は、所定の装置(図示せず)より供給されたデータを保持し、制御信号に対応してレジスタ2-2に出力する。レジスタ2-2、2-3は、レジスタ2-1、2-2よりそれぞれ供給されたデータを保持し、制御信号に対応してレジスタ2-3、2-4にそれぞれ出力する。レジスタ2-4は、レジスタ2-3より供給されたデータを保持する。

【0045】なお、レジスタ2-1乃至2-4は、直列接続されており、4段のシフトレジスタとして動作する。このシフトレジスタには、水平走査された入力画像データ系列が、ワード単位で順次入力され、連続する4つの原画像の画素データが記憶される。

【0046】また、Cubic近似法においては、このように、4段のシフトレジスタを利用して、補間点を挟む左右2画素の合計4画素のデータから補間点のデータを算出する。

【0047】乗算器3-i(i=1,...,4)は、レジスタ2-iに保持されている値と、係数メモリ1より供給された値(フィルタ係数)FCiを乗算し、その演算結果を加算器4に出力する。

【0048】加算器4は、乗算器3-1乃至3-4より供給された値の総和を演算し、補間値として出力する。

【0049】このようにして、レジスタ2-1に時系列に沿って入力されたデータとフィルタ係数は、乗算器3-1乃至3-4および加算器4において積和演算され、その演算結果である補間点のデータが時系列に沿って出力される。

【0050】次に、具体的に、原画像をCubic近似により(10/7)倍に拡大するときの、図30の演算装置の動作について説明する。

【0051】(10/7)倍の画像拡大においては、上述したように、各補間点に対する各画素の位置関係を、図28に示すように設定して、補間フィルタ演算をすべし。

【0052】図31は、各サイクルにおける、図30の演算装置の各部が有する値を示している。

【0053】なお、図30に示すようなハードウェア的に処理を行う装置では、その乗算や総和演算の部分におけるパイプライン処理により、通常、レイテンシ(高速演算実現のための遅延)が生じるが、便宜上、ここでは、レイテンシはないものと仮定する。

【0054】図31の第1サイクルにおいては、原画像

の1画素分の画像データである入力データR1が、所定の装置より供給される。このとき、レジスタ2-1乃至2-4は、入力データR1より1つ前のデータRm0、入力データR1より2つ前のデータRm1、入力データR1より3つ前のデータRm2、および、入力データR1より4つ前のデータRm3をそれぞれ保持している。

【0055】このとき、制御信号の値が「H」であるので、レジスタ2-1乃至2-4は、次のクロックの立ち上がりのタイミングで、データをそれぞれシフトする。

【0056】データがそれぞれシフトされるので、第2 10 サイクルにおいては、レジスタ2-1乃至2-4は、R1、Rm0、Rm1、および、Rm2をそれぞれ保持する。このとき、制御信号の値が「H」であるので、レジスタ2-1乃至2-4は、次のクロックの立ち上がりのタイミングで、データをそれぞれシフトする。

【0057】データがそれぞれシフトされるので、第3 サイクルにおいては、レジスタ2-1乃至2-4は、R2、R1、Rm0、および、Rm1をそれぞれ保持する。このとき、制御信号の値が「H」であるので、レジスタ2-1乃至2-4は、次のクロックの立ち上がりの 20 タイミングで、データをそれぞれシフトする。

【0058】同様に、データがそれぞれシフトされるので、第4サイクルにおいては、レジスタ2-1乃至2-4は、R3、R2、R1、および、Rm0をそれぞれ保持する。また、図23におけるRm0、R1、R2、R3に対する補間値Q1の位相を示すフィルタ選択信号P0が、係数メモリ1に供給される。このように、フィルタ選択信号は、出力信号Q1の位相Pに対応して供給される。

【0059】なお、この場合、図28のRaがRm0 30 に、RbがR1に、RcがR2に、RdがR3に対応し、図28のQが補間の値Q1に対応する。

【0060】係数メモリ1は、図32に示す10種類のフィルタ係数セットを記憶し、供給されたフィルタ選択信号Piに応じて4つの係数FC1、FC2、FC3、FC4を選択し、出力する。(10/7)倍の画像拡大の場合には、その補間演算における画素の位相は図23に示すように10種類だけであるので、係数メモリ1は、その位相の数だけのフィルタ係数セットを有している。

【0061】即ち、フィルタ選択信号Piは、図28でSを10等分する各位置に対応する10種類の位相のうちの、位相がi/10であるときのフィルタ係数セットに対応している。なお、図32の小数点表現係数(フィルタ係数)は、フィルタ選択信号Piに対応する位相をxとして式(4)に代入して算出される値であり、8ビット表現係数は、その小数点表現係数を8ビットに語長制限して(ここでは最大振幅を128とした)算出される。

【0062】今の場合、係数メモリ1は、フィルタ選択 50

信号がP0であるので、図32の位相P0に対応するフィルタ係数セット(0.0, 1.0, 0.0, 0.0, 0.0)(8ビット表現の場合、(0, 128, 0, 0))を、4つのフィルタ係数FC1、FC2、FC3、FC4として乗算器3-1乃至3-4にそれぞれ出力する。

【0063】そして、乗算器3-1乃至3-4および加算器4により、上述の積和演算が行われ、その演算結果が、出力データQ1として出力される。

【0064】このとき(第4サイクルにおいて)、制御信号の値は「L」であるので、次のクロックにおいて、レジスタ2-1乃至2-4は、保持しているデータのシフトを行わない。

【0065】第5サイクルにおいては、レジスタ2-1乃至2-4は、第4サイクルにおいて保持していたデータを、継続して保持する。このとき、図23におけるRm0、R1、R2、R3に対する出力データQ2の位相を示すフィルタ選択信号P7が、係数メモリ1に供給される。

【0066】なお、この場合、図23に示すように、R1とQ1が同位相であり、かつ、Q1とQ2との間隔が、R1とR2との間隔Sの7/10であるので、位相は7/10となり、フィルタ選択信号P7が供給される。

【0067】そして、係数メモリ1は、フィルタ選択信号がP7であるので、図32の7/10の位相に対応するフィルタ係数セット(-0.147, 0.363, 0.847, -0.063)(8ビット表現の場合、(-19, 46, 108, -8))を、4つのフィルタ係数FC1、FC2、FC3、FC4として乗算器3-1乃至3-4にそれぞれ出力する。

【0068】そして、乗算器3-1乃至3-4および加算器4により、上述の積和演算が行われ、その演算結果が、出力データQ2として出力される。

【0069】なお、この場合のように、第4サイクルの出力データQ1と第5サイクルの出力データQ2の算出に使用される原画像の画素が同一である場合、第4サイクルにおける制御信号の値を「L」として、第4サイクルから第5サイクルに移行する時にレジスタ2-1乃至2-4をシフト動作させない。

40 【0070】このとき(第5サイクルにおいて)、制御信号の値は「H」であるので、レジスタ2-1乃至2-4は、次のクロックの立ち上がりのタイミングで、データをそれぞれシフトする。

【0071】次に、第6サイクルにおいては、レジスタ2-1乃至2-4は、R4、R3、R2、および、R1をそれぞれ保持する。また、図23におけるR4、R3、R2、R1に対する出力データQ3の位相を示すフィルタ選択信号P4が、係数メモリ1に供給される。

【0072】なお、今回の位相は、前回の位相7/10 50 に、7/10が累加されて14/10となるが、原画像

データ1個分に相当する位相(=10/10)を減算するので、今回の位相は、 $4/10$ (= $7/10+7/10-10/10$)となる。

【0073】即ち、1回のサイクル毎に、位相は、本来、 $7/10$ ずつ変化する。そして、整数分はデータ遅延として扱われるので、結局、位相は、モジュロ演算されることになる。

【0074】係数メモリ1は、フィルタ選択信号がP4であるので、図32のP4に対応するフィルタ係数セット(−0.096, 0.744, 0.496, −0.144)(8ビット表現の場合、(−12, 95, 63, −18))を、4つのフィルタ係数FC1, FC2, FC3, FC4として乗算器3-1乃至3-4に出力する。

【0075】そして、乗算器3-1乃至3-4および加算器4により、上述の積和演算が行われ、その演算結果が、出力データQ3として出力される。

【0076】このとき、制御信号の値が「H」であるので、レジスタ2-1乃至2-4は、次のクロックの立ち上がりタイミングで、データをそれぞれシフトする。

【0077】以下同様にして、図31に示すように、処理が進められていき、出力データQiが、順次出力される。

【0078】なお、上述の装置を利用して画像の画素数変換などを施した場合、その入力のデータレートと出力のデータレートは画素数の変化に起因して変化してしまう。

【0079】例えば、上述のように画素数が増える変換においては、出力データのレートは一定になっているものの、入力データ列の供給は、図31の第5サイクルのように、停止することがある。また、画素数が減る変換の場合においては、入力は一定であるものの、出力データの出力が停止することがある。

【0080】従って、実際には、図30に示す演算装置の入出力データを一旦記憶するバッファメモリを設けることにより、データレートを一定に保つようにしている。

【0081】以上のようにして、(水平方向における)画像の拡大または縮小、および、解像度の変換が、ハードウェア的に(即ち、処理における各演算に対応して構成されている電子回路を利用して)行われている。

【0082】しかしながら、上述のようにハードウェア的に、画像の拡大または縮小、および、解像度の変換を行う装置を利用する場合、画像の変換時において、同時に行うことが所望される、各種画像処理、テレビジョン信号処理、ノイズ除去などを行うために、別途、各処理に対応する装置が必要となる。

【0083】そこで、例えば、先に提案した特願平7-246627号に記載されているように、SIMD(Single Instruction Multiple Data stream)形式の並列

プロセッサを利用して、ソフトウェア的に、上述の演算を行う方法が考えられている。

【0084】図33は、そのような並列プロセッサの構成例を示している。この並列プロセッサは、入力ポインタ21、入力SAM(シリアルアクセスメモリ)部22、データメモリ部23、ALUアレイ部24、出力SAM部25、出力ポインタ26、および、プログラム制御部27で構成されている。

【0085】入力SAM部22、データメモリ部23、ALUアレイ部24、および、出力SAM部25は、リニアアレイ(直線配列)型に並列化された要素プロセッサ群を構成している。これらの要素プロセッサ31は、プログラム制御部27が有する1つのプログラムに従って、連動して制御される(即ち、SIMD制御される)。プログラム制御部27は、プログラムメモリや、そのプログラムを進行させるシーケンス制御回路などを有し、プログラムメモリに予め書き込まれたプログラムに従って、各種制御信号を発生して、各種回路を制御する。

【0086】なお、入力SAM部22、データメモリ部23、出力SAM部25は、主にメモリで構成されている。詳細に説明しないが、図33の装置においては、これらのメモリのための「ロウ(ROW)」アドレスデコーダは、プログラム制御部27に含まれているものとする。

【0087】並列化された要素プロセッサ31(単一エレメント分)は、図33において、斜線で示した部分に対応し、複数の要素プロセッサ31が、図中において横方向に配列されている。即ち、図33の斜線の部分だけで、1つのプロセッサに対応する構成要素を有している。

【0088】次に、図33の画像処理用のリニアアレイ型並列プロセッサの動作について説明する。

【0089】入力端子DINに与えられた入力データ(1画素分の画像データ)は、入力SAM部22に供給される。

【0090】入力ポインタ21は、1つの入力データに対して1つの要素プロセッサ31だけに、値「H」の1ビット信号、即ち入力ポインタ信号(SIP)を出力する。そして、値「H」で指定された要素プロセッサ31の入力SAM部22(入力SAMセル)に、その入力データが書き込まれる。

【0091】入力ポインタ信号による、データを供給する要素プロセッサ31の指定は、入力データのクロックごとに図中の左端の要素プロセッサ31から右端の要素プロセッサ31に向けて順次移動するので、入力データは、左端の要素プロセッサ31の入力SAM部22(入力SAMセル)から、右側の要素プロセッサ31の入力SAMセルに順次供給される。

【0092】要素プロセッサ31の数は画像信号の1水

水平走査期間の画素数H以上に設計されているので、画像信号の1水平走査期間分の画素データを、入力SAM部22に蓄積することができる。このような入力動作は、水平走査期間毎に繰り返される。

【0093】プログラム制御部27は、このようにして画像信号の1水平走査期間のデータが入力SAM部22に蓄積されるごとに、プログラムに従って入力SAM部22、データメモリ部23、ALUアレイ部24、および、出力SAM部25を以下のようにSIMD制御して処理を実行する。

【0094】なお、このプログラム制御は水平走査期間ごとに繰り返される。従って、水平走査期間に対応する時間を、このプロセッサの命令サイクル周期で割算して算出されるステップ数だけのプログラムを処理することができる。SIMD制御であるから、以下の動作は全ての要素プロセッサ31において並行して同様に実行される。

【0095】入力SAM部22に蓄積された1水平走査期間分の入力データは、次の水平走査期間において、必要に応じて入力SAM部22からデータメモリ部23へ転送され、その後の演算処理に使われる。

【0096】入力SAM部22からデータメモリ部23へのデータの転送においては、プログラム制御部27は、入力SAM読み出し信号(SIR)により入力SAM部22の所定のビットのデータを選択してアクセスした後、メモリアクセス信号(SWA)を出して、そのデータを、データメモリ部23の所定のメモリセル(後述)へ書き込んでいく。

【0097】次に、プログラム制御部27は、プログラムに応じて、各要素プロセッサ31のデータメモリ部23に保持されているデータを、その要素プロセッサ31のALUアレイ部24に供給し、そのデータに対して算術演算あるいは論理演算を行わせる。そして、その演算結果は、データメモリ部23の所定のアドレスに書き込まれる。

【0098】なお、ALUアレイ部24における演算は、全てビット単位で行われるので、1サイクル当たり1ビットずつ処理が進行する。例えば、8ビットのデータ同士の論理演算を行う場合、少なくとも8サイクルかかることになる。また、8ビットのデータ同士の加算を行う場合、少なくとも9サイクルかかることになる。8ビットのデータ同士の乗算を行う場合、その乗算は64回のビット加算と等価であるので、少なくとも64サイクルかかることになる。

【0099】また、要素プロセッサ31は、近傍の要素プロセッサ31に接続されており、プロセッサ間通信を行うことができる。ただし、近傍の要素プロセッサ31のデータメモリ部23へのアクセスを行う場合、SIMD制御に起因して、例えば右隣りの要素プロセッサ31のデータメモリ部をアクセスするときは、すべての要素

プロセッサ31が、右隣りの要素プロセッサ31のデータメモリ部23をアクセスすることになる。

【0100】なお、このように動作することは、FIRデジタルフィルタの実現には特に問題とはならない。また、直接接続されていない要素プロセッサ31のデータを読み出す場合、プログラムステップは多少増えるが、近傍のプロセッサ間通信を繰り返すことにより、データを読み出す。

【0101】このような通信を利用して、近傍の要素プロセッサ31が保持するデータを利用して、画像の水平方向のFIRデジタルフィルタ演算を実現することができる。

【0102】なお、このような並列プロセッサにおいては、画面の水平方向の同じ位置の画素のデータは、すべての水平走査期間において、所定の1つの要素プロセッサ31により処理されるので、データを入力SAM部22からデータメモリ部23に転送するときに、水平走査期間ごとに、データを記憶するアドレスを変更することにより、過去の水平走査期間の入力データを、それ以降の水平走査期間まで、データメモリ部23の内部に保持することができる。このようにすることにより、画像の垂直方向のFIRデジタルフィルタについても、その演算に必要なデータをデータメモリ部23に順次保持させていくことができる。

【0103】このようにして、それぞれの要素プロセッサ31は、垂直方向(水平走査方向に対して垂直な方向)の所定の連続数の画素データを、データメモリ部23の内部に保持し、垂直方向のFIRデジタルフィルタ演算を実現している。

【0104】以上のようにして1水平走査期間に割り当てられている演算が終了すると、その水平走査期間のうちに、その水平走査期間に演算したデータは、出力SAM部25に転送される。

【0105】このように、1水平走査期間のうちに、入力SAM部22に蓄積された入力データのデータメモリ部23への転送、ALUアレイ部24による演算、および、出力SAM部25へのデータの転送が、ビットを単位とするSIMD制御プログラムに従って実行される。そして、これらの処理は、水平走査期間を単位として、順次繰り返される。

【0106】そして、出力SAM部25に転送された出力データは、さらに次の水平走査期間において、出力SAM部25から出力される。

【0107】以上のように、入力データを入力SAM部22に書き込む入力処理、プログラム制御部27による、入力SAM部22に蓄積された入力データのデータメモリ部23への転送、ALUアレイ部24による演算、および、出力SAM部25への出力データの転送の演算処理、並びに、出力データを出力SAM部25から出力させる出力処理の3つの処理が、各水平走査入力デ

ータに対して行われる。なお、これらの3つの処理は、画像信号の1水平走査期間を単位とするパイプライン処理として実行される。

【0108】1つの水平走査期間の入力データに注目すれば、その入力データに対する3つの処理には、各処理において1水平走査期間に対応する時間が経過するので、合計水平走査期間の3倍に対応する時間がかかるが、3つの処理がパイプライン処理として並行に実行されるので、平均して、1水平走査期間分の入力データあたり1水平走査期間に対応する時間で処理を行うことができる。

【0109】

【発明が解決しようとする課題】しかしながら、上述の装置においては、所定の画像の変換比率を $K:L$ とすると、変換後の画素数である L 個のフィルタ係数セットが必要であり、変換比率が簡単な整数比ではない場合、多くのフィルタ係数セットが必要となり、フィルタ係数セットを記憶する記憶部に大容量のものを使用することになるため、装置のコストおよび回路規模を低減することが困難であるという問題を有している。

【0110】また、例えばユーザによる操作などに応じて、変換比率を変化させるようにする場合、複数の変換比率に対応するフィルタ係数セットを使用することが考えられるが、その場合、変換比率の数と変換後の画素数の積に対応する数のフィルタ係数セットが必要となり、大容量の記憶部が必要となるので、装置のコストおよび回路規模を低減することが困難であるという問題を有している。

【0111】さらに、上述の並列プロセッサのように一般的なFIRデジタルフィルタの実現は可能であるが、補間の演算が必要とされる画像の拡大や縮小を行う場合（同様に、解像度の変換を行う場合）、補間の演算はFIRデジタルフィルタの一種ではあるものの、入力SAM部22に保持されるデータの数と、出力SAM部25に出力されるデータの数が異なるので、入力SAM部22または出力SAM部25において、入力データ R_i または出力データ Q_i は、密に配列されない。

【0112】従って、要素プロセッサ31が、補間の処理に必要な入力画素データを、所定の数の他の要素プロセッサ31より獲得する場合、その要素プロセッサ31と、他の要素プロセッサ31との位置関係が、要素プロセッサ31毎に異なるので、すべての要素プロセッサ31が同様の動作を行うSIMD形式の並列プロセッサでは、必要なデータを獲得することが困難であるという問題を有している。

【0113】例えば、Cubic近似では連続する入力データのうちの4つのデータに対する畳み込み演算が必要である。例えば、画像を $(10/7)$ 倍に拡大する場合、図34に示すように、入力データ R_i は、密に配列されないのので、例えば出力データ Q_3 を算出するときに

必要となる入力データ R_1, R_2, R_3, R_4 のうち、 R_1, R_3, R_4 は、出力データ Q_3 を算出する要素プロセッサ31を基点として、左2つ隣り、右1つ隣り、および、右3つ隣りの要素プロセッサ31によってそれぞれ保持されている。

【0114】一方、出力データ Q_4 の算出に必要な入力データ R_2, R_3, R_4, R_5 のうち、 R_2, R_4, R_5 は、左1つ隣り、右2つ隣り、および、右3つ隣りの要素プロセッサ31によってそれぞれ保持されている。また、出力データ Q_5 の算出に必要な入力データ R_2, R_3, R_4, R_5 は、基点となる要素プロセッサ31の左2つ隣り、左1つ隣り、右1つ隣り、右2つ隣りの要素プロセッサ31にそれぞれ保持されている。

【0115】このように、各出力データの算出に必要な入力データを保持している要素プロセッサ31と、その出力データを算出する要素プロセッサ31との位置関係が、出力データ毎に変化する。

【0116】図35は、画像を $(10/7)$ 倍に拡大する場合の、各出力データの算出に必要な入力データを保持している要素プロセッサ31のパターンを示している。図35に示すように、この場合、5つのパターンに分類される。

【0117】また、上述のような並列プロセッサを利用する場合、各要素プロセッサ31が、1画素に対応する出力データを算出するので、要素プロセッサ31毎に、異なる上述のフィルタ係数を供給する必要があるという問題を有している。

【0118】本発明は、このような状況に鑑みてなされたもので、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、補間演算される画素の位相に最も近いフィルタ係数セットを利用して、画像データの補間演算を行うようにして、任意の変換比率の画像の拡大または縮小を行うことができるようにするものである。

【0119】

【課題を解決するための手段】請求項1に記載の画像処理装置は、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを記憶し、そのフィルタ係数セットのうち、位相が、補間演算される画素データの位相に最も近いフィルタ係数セットを演算部に出力する記憶部と、そのフィルタ係数セットを利用して、画素データの補間演算を行う演算部とを備えることを特徴とする。

【0120】請求項3に記載の画像処理方法は、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを記憶している記憶部から、位相が、補間演算される画素データの位相に最も近いフィルタ係数セットを演算部に出力し、演算部により、フィルタ係数セットを利用して、画素データの補間演算を行うことを特徴とする。

【0121】請求項4に記載の画像処理装置は、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、処理される画素データの位相に最も近いフィルタ係数セットを、要素プロセッサにそれぞれ供給し、要素プロセッサが、フィルタ係数セットを利用して、画素データの補間の処理をそれぞれ行うことを特徴とする。

【0122】請求項19に記載の画像処理方法は、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、処理される画素データの位相に最も近いフィルタ係数セットを、要素プロセッサにそれぞれ供給し、要素プロセッサが、フィルタ係数セットを利用して、画素データの補間の処理をそれぞれ行うことを特徴とする。

【0123】請求項1に記載の画像処理装置において、記憶部は、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、補間演算される画素データの位相に最も近いフィルタ係数セットを演算部に出力し、演算部は、そのフィルタ係数セットを利用して、画素データの補間演算を行う。

【0124】請求項3に記載の画像処理方法において、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを記憶している記憶部から、位相が、補間演算される画素データの位相に最も近いフィルタ係数セットを演算部に出力し、演算部により、フィルタ係数セットを利用して、画素データの補間演算を行う。

【0125】請求項4に記載の画像処理装置において、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、処理される画素データの位相に最も近いフィルタ係数セットを、要素プロセッサにそれぞれ供給し、要素プロセッサが、フィルタ係数セットを利用して、画素データの補間の処理をそれぞれ行う。

【0126】請求項19に記載の画像処理方法において、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、処理される画素データの位相に最も近いフィルタ係数セットを、要素プロセッサにそれぞれ供給し、要素プロセッサが、フィルタ係数セットを利用して、画素データの補間の処理をそれぞれ行う。

【0127】

【発明の実施の形態】図1は、本発明の画像処理装置の第1の実施の形態の構成例を示している。

【0128】係数メモリ1Aは、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを記憶するようになっている。

【0129】例えば、分割数を16に設定した場合、係数メモリ1Aは、図2に示すように、正規化位相量 x お

よびフィルタ選択信号 P_i に対応する16個のフィルタ係数セット(FC1, FC2, FC3, FC4)を予め記憶している。

【0130】制御回路5は、画像の変換に伴う、補間後の画素間隔に対応する位相の変化分 P_d を所定の装置(図示せず)より供給され、その位相変化分 P_d に対応して、補間する画素に対応するフィルタ係数セットが選択されるようにフィルタ選択信号 P_i を発生して、そのフィルタ選択信号 P_i を係数メモリ1Aに出力するようになっている。

【0131】また、制御回路5は、位相変化分 P_d の累積値に対応して、レジスタ2-1乃至2-4に供給される制御信号を生成する。

【0132】制御回路5の剰余回路11は、レジスタ12に記憶されている値と、位相変化分 P_d との和を1で割算したときの剰余(即ち、和の小数部分)をレジスタ12に出力するようになっている。また、剰余回路11は、レジスタ12に記憶されている値と、位相変化分 P_d との和が1以上である場合、所定の信号を制御信号発生回路14および調整回路15に出力するようになっている。

【0133】制御回路5のレジスタ12は、剰余回路11より供給された値を保持し、その値を剰余回路11および近似回路13に出力するようになっている。また、レジスタ12は、1水平走査期間毎に供給されるクリア信号に対応して、保持する値を0にリセットする。

【0134】制御回路5の近似回路13は、レジスタ12より供給された値に最も近い正規化位相量 x (図2)に対応するフィルタ選択信号 P_i を係数メモリ1Aに出力するようになっている。

【0135】制御回路5の制御信号発生回路14は、剰余回路11より所定の信号が供給された場合、値が「L」の制御信号をレジスタ2-1乃至2-4に出力し、剰余回路11より所定の信号が供給されていない場合、値が「H」の制御信号を出力するようになっている。

【0136】制御回路の調整回路15は、バッファメモリ(図示せず)を内蔵し、供給される入力画素データを、所定のタイミングでレジスタ2-1に出力するようになっている。調整回路15は、剰余回路11より所定の信号が供給された場合、そのサイクルにおける画素データの出力を停止する。

【0137】なお、レジスタ2-1乃至2-4、乗算器3-1乃至3-4、および加算器4は、図30と同様に構成されているので、その説明を省略する。

【0138】次に、図1の画像処理装置の動作について説明する。

【0139】最初に、位相変化分 P_d が制御回路5の剰余回路11に供給される。そして、剰余回路11は、供給された値と、レジスタ12の値の和の小数部分をレジ

スタ12に出力する。なお、このとき、供給された値と、レジスタ12の値の和が1以上である場合、剰余回路11は、所定の信号を制御信号発生回路14および調整回路15に出力する。

【0140】そして、近似回路13は、位相 x が、レジスタ12の値に最も近い位相に対応するフィルタ係数セットに対応するフィルタ選択信号 P_i を係数メモリ1Aに出力する。

【0141】このようにして、所定の画素データの補間において、所定の数のフィルタ係数セットのうちの最適なフィルタ係数セットが選択される。

【0142】なお、そのフィルタ係数セットを利用して補間値を算出する動作は、図30の装置と同様であるので、その説明を省略する。

【0143】次に、図3を参照して、例えば、予め係数メモリ1Aに記憶されている16個のフィルタ係数セットを利用して、 $(10/7)$ 倍に画像を拡大するときの制御回路5の動作を説明する。

【0144】最初に、位相が0である第1番目の補間値 Q_1 （図23）を算出する場合においては、レジスタ12の値は初期値0に設定されており、その値が近似回路13に供給され、近似回路13は、図2に示すフィルタ選択信号 P_i のうち、位相 x が、供給された値0に最も近いフィルタ選択信号 P_0 （ $x=0.0$ ）を選択し、そのフィルタ選択信号 P_0 を係数メモリ1Aに出力する。なお、このとき、フィルタ選択信号 P_0 に対応する位相 x が0.0であり、供給された値と同一であるので、位相に関する誤差は0である。

【0145】また、このとき、剰余回路11には、位相変化分 P_d （この場合、 $P_d=0.7$ ）が供給され、レジスタ12の値0と位相変化分0.7の和の小数部分、即ち、0.7が、レジスタ12に出力される。

【0146】次に、位相が0.7である第2番目の補間値 Q_2 （図23）を算出する場合、このときのレジスタ12の値は0.7に設定されており、その値が近似回路13に供給され、近似回路13は、図2に示すフィルタ選択信号 P_i のうち、位相 x が、供給された値0.7に最も近いフィルタ選択信号 P_{11} （ $x=0.6875$ ）を選択し、そのフィルタ選択信号 P_{11} を係数メモリ1Aに出力する。なお、このとき、フィルタ選択信号 P_{11} に対応する位相 x が0.6875であり、供給された値0.7と異なるので、0.0125の位相に関する誤差が発生している。

【0147】また、このとき、剰余回路11には、位相変化分 P_d （ $P_d=0.7$ ）が供給され、レジスタ12の値0.7と位相変化分0.7の和の小数部分、即ち、0.4が、レジスタ12に出力される。

【0148】そして、位相が0.4である第3番目の補間値 Q_3 （図23）を算出する場合、このときのレジスタ12の値は0.4に設定されており、その値が近似回

路13に供給され、近似回路13は、図2に示すフィルタ選択信号 P_i のうち、位相 x が、供給された値0.4に最も近いフィルタ選択信号 P_6 （ $x=0.375$ ）を選択し、そのフィルタ選択信号 P_6 を係数メモリ1Aに出力する。なお、このとき、フィルタ選択信号 P_6 に対応する位相 x が0.375であり、供給された値0.4と異なるので、0.025の位相に関する誤差が発生している。

【0149】また、このとき、剰余回路11には、位相変化分 P_d （ $P_d=0.7$ ）が供給され、レジスタ12の値0.4と位相変化分0.7の和の小数部分、即ち、0.1が、レジスタ12に出力される。

【0150】以下同様にして、第4番目乃至第10番目の補間値 Q_i が算出されていく。第11番目の補間値 Q_{11} を算出するとき、補間する画素の位相が0に戻るのので、この場合においては、16個のフィルタ係数セットのうち、合計10個のフィルタ係数セットが利用される。

【0151】位相変化分 P_d は変換比率 $K:L$ の L によって決まるので、上述の位相変化分 P_d の値を変更するだけで、異なる変換比率の画像の拡大または縮小を行うことができる。

【0152】なお、このように変換比率に拘わらず、所定の数のフィルタ係数セットを利用する場合、上述のように位相の誤差が発生するが、元々、補間関数として、 sinc 関数を大幅に近似したものを利用しているので、この程度の位相の誤差は、ほとんど問題とはならない。なお、必要に応じて、画素間隔の分割数を大きくしてもよい。

【0153】以上のように、第1の実施の形態においては、任意の変換比率で、ハードウェア的に画像の拡大または縮小を行うことができる。

【0154】なお、上述の説明においては、画素間隔の分割数が16に設定されているが、勿論、他の分割数でもよい。

【0155】また、上述の位相変化分 P_d は、小数で表現されているが、画素間隔を分割数で除算して得られる単位位相量で、その小数を除算した整数値を位相変化分としてもよい。その場合、分割数を2のべき乗にすることにより、剰余回路11による剰余を算出する処理が簡単になる。即ち、その場合、剰余回路11は、上位ビットを無視するようにした2進数の加算器で実現することができる。また、その場合、近似回路13は、単なる語長制限のため四捨五入を行う回路でよい。

【0156】さらに、上述のように、分割数に対応する数（今の場合、16個）のフィルタ係数セットのうち、所定の数（今の場合、10個）のフィルタ係数セットだけが使用される場合、係数メモリ1Aが、使用されるフィルタ係数セットだけを記憶するようにしてもよい。その場合、係数メモリ1Aの記憶領域を節約することがで

きる。

【0157】図4は、本発明の画像処理装置の第2の実施の形態の構成を示している。

【0158】入力ポインタ21は、要素プロセッサ31毎に、入力SAM部22に、入力データを受け取るか否かを示す入力ポインタ信号を出力するようになされている。なお、この入力ポインタ21は、例えば特開平8-123683号公報に記載されているものと同様に、入力データを選択的に要素プロセッサ31に供給させることができる。

【0159】入力SAM部22は、要素プロセッサ31毎に、所定の入力データを保持する記憶部を有し、入力ポインタ21より供給されるSIP信号に対応して、入力データを記憶するようになされている。また、入力SAM部22は、プログラム制御部27Aより、SIR信号を受け取ると、保持しているデータを、データメモリ部23に出力するようになされている。

【0160】データメモリ部23は、要素プロセッサ31毎に、所定のデータを保持する記憶部を有し、プログラム制御部27Aより、SWA信号を受け取ると、入力SAM部22またはALUアレイ部24より供給されたデータを記憶部に記憶し、メモリ読み出しアクセス信号(SRAA, SRBA)を受け取ると、そのデータをALUアレイ部24に出力するようになされている。

【0161】ALUアレイ部24は、要素プロセッサ31毎に演算部(図6のALU(Arithmetic and Logical Unit)81)を有し、データメモリ部23より供給されるデータに対して、プログラム制御部27Aより供給されるALU制御信号(SALU-CONT)に対応する演算を行うようになされている。

【0162】出力SAM部25は、要素プロセッサ31毎に、所定の出力データを保持する記憶部を有し、プログラム制御部27Aより供給される出力SAM書き込み信号(SOW)を受け取ると、ALUアレイ部24からの出力データをその記憶部に記憶するようになされている。また、出力SAM部25は、出力ポインタ26より供給される出力ポインタ信号(SOP)に対応して、保持しているデータを出力するようになされている。

【0163】出力ポインタ26は、要素プロセッサ31毎に、出力SAM部25に、出力データを出力するか否かを示すSOP信号を出力するようになされている。なお、この出力ポインタ26は、例えば特開平8-123683号公報に記載されているものと同様に、要素プロセッサ31から選択的にデータを出力させることができる。

【0164】プログラム制御部27Aは、所定のプログラムに従って各部を制御し、後述の各種動作を行わせるようになされている。

【0165】図5は、要素プロセッサ31の構成例を示している。図5の要素プロセッサ31は、様々な用途に

利用できる汎用的なプロセッサ形態として構成されている。入力バッファメモリ(IQ)41は、図4の入力SAM部22の1要素プロセッサ分に対応し、入力データを記憶する。データメモリ(RF)42は、図4のデータメモリ部23の1要素プロセッサ分に対応し、演算途中のデータなどを記憶する3ポートメモリである。出力バッファメモリ(OQ)44は、図4の出力SAM部25の1要素プロセッサ分に対応し、出力データを記憶する。

10 【0166】演算部(ALU)43は、図4のALUアレイ部24の1要素プロセッサ分に対応し、データメモリ42より供給されるデータに対して各種演算を行い、その演算結果をデータメモリ42または出力バッファメモリ44に出力する。

【0167】図5の要素プロセッサ31においては、入力データは、入力バッファメモリ41に一旦入力され、その後、データメモリ42に転送される。演算部43は、データメモリ42より必要に応じて供給される、新たに記憶されたデータ、過去に記憶されたデータ、演算途中のデータなどに対して各種演算を行い、再びデータメモリ42に書き込むという作業を、プログラムに対応して繰り返す。その演算結果は、出力バッファメモリ44に転送され、所定の速度やフォーマットで出力される。

【0168】なお、要素プロセッサ31においては、入力SAM部22、データメモリ部23、および、出力SAM部25が、メモリの「カラム」を構成している。また、ALUアレイ部24は、1ビットALUであり、フルアダー(全加算器)を主体にした回路構成を有している。従って、普通の所謂パーソナルコンピュータなどに利用されているワード単位に処理を行うプロセッサとは異なり、この要素プロセッサ31は、ビット処理プロセッサであり、ビットを単位として処理を行う。

【0169】ビット処理プロセッサは、1プロセッサあたりのハードウェアの規模が小さいので、並列数を大きくすることができる。従って、画像処理用の並列プロセッサは、要素プロセッサ31の直線配列の並列数、画像信号の一水平走査期間分の画素数Hと同一もしくはそれより多く設計されている。

40 【0170】図6は、上述の要素プロセッサ31の詳細な回路構成の一例を示している。なお、図6の各セルの構造は、理解を容易にするために非常に一般的なものとして記述されている。また、同じ回路が複数並ぶ部分は、1つの回路(1ビット分の回路)で代表して記述されている。

【0171】入力SAM部22の1つの要素プロセッサ31に対応する部分は、入力ポインタ21に制御され、入力データのビット数ISBに対応して、ISB個の、1ビットを記憶する入力SAMセル22-1乃至22-ISBで構成されている。なお、図6においては、IS

B個の入力SAMセル22-1乃至22-ISBの代わりに、1つのセル22-iが記載されている。

【0172】入力SAMセル22-iにおいては、トランジスタTr1のゲート端子は、入力ポインタ21に接続され、トランジスタTr1の他の2つの端子は、入力データバス65または1ビットのデータを記憶するコンデンサC1の一端に接続されている。

【0173】また、トランジスタTr2のゲート端子は、プログラム制御部27Aに接続されており、SIR信号を供給され、トランジスタTr2の他の2つの端子は、書き込みビット線63およびコンデンサC1の一端にそれぞれ接続されている。

【0174】コンデンサC1の一端は、トランジスタTr1、Tr2に接続され、他端は、接地されている。

【0175】データメモリ部23の1つの要素プロセッサ31に対応する部分は、作業メモリとして必要なビット数MBに対応するMB個のデータメモリセル23-1乃至23-MBで構成されている。なお、図6においては、MB個のデータメモリセル23-1乃至23-MBの代わりに、1つのセル23-iが記載されている。

【0176】データメモリ部23のデータメモリセル23-i (i=1, ..., MB) は、2本の読み出しビット線61、62と1本の書き込みビット線63を有する3ポートメモリである。

【0177】データメモリセル23-iにおいては、トランジスタTr11のゲート端子は、プログラム制御部27Aに接続されており、SWA信号を供給され、トランジスタTr11の他の2つの端子は、書き込みビット線63および1ビットのデータを記憶するコンデンサC11の一端にそれぞれ接続されている。

【0178】コンデンサC11の一端は、トランジスタTr12のゲート端子と、トランジスタTr11に接続され、他端は、接地されている。

【0179】トランジスタTr12の残りの2つの端子は、接地点、および、抵抗Rを介して電源（図示せず）にそれぞれ接続されている。なお、抵抗Rは省略してもよい。

【0180】トランジスタTr13のゲート端子は、プログラム制御部27Aに接続されており、SRAA信号を供給され、トランジスタTr13の残りの2つの端子は、トランジスタTr12と抵抗R、および、読み出しビット線61にそれぞれ接続されている。

【0181】トランジスタTr14のゲート端子は、プログラム制御部27Aに接続されており、SRBA信号を供給され、トランジスタTr14の残りの2つの端子は、トランジスタTr12と抵抗R、および、読み出しビット線62にそれぞれ接続されている。

【0182】ALUアレイ部24の1つの要素プロセッサ31に対応する部分は、図6におけるALUセル24A (ALU部) である。ALUセル24AのALU81

は、1ビットALUであり、全加算器（フルアダー）などの回路構成を有し、フリップフロップ82-1乃至82-3より供給される1ビットの値に対して演算を行い、その演算結果をセクタ83に出力する。

【0183】また、ALUセル24Aは、ALU81に入力される1ビットの値を保持するフリップフロップ82-1乃至82-3、フリップフロップ82-1乃至82-3に供給される値を選択するセクタ（SEL）84-1乃至84-3などを有している。

【0184】出力SAM部25の1つの要素プロセッサ31に対応する部分は、出力ポインタ26に制御され、出力信号ビット数（OSB）に対応して、OSB個の出力SAMセル25-1乃至25-OSBで構成されている。なお、図6においては、出力SAMセル25-1乃至25-OSBの代わりに、1つのセル25-iが記載されている。

【0185】出力SAMセル25-iにおいては、トランジスタTr7のゲート端子は、プログラム制御部27Aに接続されており、SOW信号を供給され、トランジスタTr7の他の2つの端子は、書き込みビット線63A、および、1ビットのデータを記憶するコンデンサC4の一端にそれぞれ接続されている。

【0186】コンデンサC4の一端は、トランジスタTr7、Tr8に接続され、他端は、接地されている。

【0187】トランジスタTr8のゲート端子は、出力ポインタ26に接続され、トランジスタTr8の他の2つの端子のうちの一端は、コンデンサC4とトランジスタTr7に接続され、残りの一端は、出力データバス66に接続されている。

【0188】要素プロセッサ31に接続されているすべてのワード線は、それぞれ配列されている他の要素プロセッサ31にも接続されており、SIR信号、SWA信号、メモリ読み出しアクセス信号（SRAA, SRBA）、SOW信号などをすべての要素プロセッサ31に伝送する。なお、これらのワード線は、図4のプログラム制御部27A内でアドレスデコードされている。

【0189】また、入力データバス65は、すべての要素プロセッサ31の入力SAMセル22-iに接続され、出力データバス66は、すべての要素プロセッサ31の出力SAMセル25-iに接続されている。

【0190】また、図4のメモリ28（記憶手段）は、起動時、水平帰線期間、垂直帰線期間などに外部の制御用CPU（図示せず）などより供給される、すべての要素プロセッサ31におけるフィルタ演算に必要なすべての補間フィルタ係数のデータを、要素プロセッサ31の番号順に保持するようになされている。即ち、メモリ28は、第1の実施の形態の係数メモリ1Aと同様に、予め設定された分割数だけのフィルタ係数セットを記憶している。

【0191】次に、この要素プロセッサ31におけるデ

ータの転送および演算について説明する。

【0192】入力ポインタ21により指定された要素プロセッサ31の入力SAMセル22-iにおいては、トランジスタTr1がオン状態になり、コンデンサC1の端子電圧が、入力データバス65（およびバッファ71）を介して供給される入力データに応じた電圧になる。

【0193】このようにして、入力データが、指定された要素プロセッサ31の入力SAM部22に記憶される。

【0194】次に、プログラム制御部27Aより供給されるSIR信号により選択された入力SAMセル22-iにおいては、トランジスタTr2がオン状態となり、コンデンサC1の電圧に応じた転送データ信号が書き込みビット線63に生じる。

【0195】このとき、SBC信号がバッファ72に供給されるとともに、SWA信号が、所定のデータメモリセル23-iのトランジスタTr11に供給され、トランジスタTr11がオン状態になることにより、コンデンサC11の端子電圧が、入力SAMセル22-iのコンデンサC1に記憶されているデータに応じた電圧になる。

【0196】なお、ALUセル24Aからのデータをデータメモリセル23-iに書き込む場合は、SBCA信号が、バッファ73に供給される。

【0197】このデータ転送は、書き込みビット線63を介して、1サイクルに1ビットずつ行われる。入力SAM部22の各入力SAMセル22-iからデータを読み出すときに利用されるSIR信号と、データメモリ部23の各データメモリセル23-iへのデータの書き込み40に利用されるSWA信号は、同じアドレス空間内のアドレスを示しており、それぞれロウデコーダでデコードされてワード線として与えられている。

【0198】そして、ALUセル24Aは、データメモリ部23に、上述のようにして書き込まれた入力データや演算途中のデータ、あるいは、フリップフロップ82-1乃至82-3に記憶されているデータを用いて、ビット単位の演算処理を順次進める。

【0199】例えば、データメモリ部23の所定のビットに対応するデータメモリセル23-iのデータと、他のビットに対応するデータメモリセル23-iのデータを加算して、さらに他のビットに対応するデータメモリセル23-iに加算結果を書き込む場合は、次のように動作する。

【0200】プログラム制御部27Aは、データメモリ部23の所定のビットに対応するデータメモリセル23-iにSRAA信号を供給し、そのセルのトランジスタTr13をオン状態にして、コンデンサC11に記憶されていたデータを一方の読み出しビット線61に出力させる。

【0201】同時に、プログラム制御部27Aは、他のビットに対応するデータメモリセル23-iにSRBA信号を供給し、そのセルのトランジスタTr14をオン状態にして、コンデンサC11に記憶されていたデータを、他方の読み出しビット線62に出力される。

【0202】これら2つの読み出されたデータは、ALUセル24Aのセクタ84-1乃至84-3を介してALU81に供給される。そして、ALU81は、それらのデータに対して所定の演算を行い、その演算結果を、セクタ83を介してフリップフロップ85に供給する。

【0203】そして、プログラム制御部27Aは、SBCA信号を供給して、フリップフロップ85から演算結果を書き込みビット線63に出力させ、SWA信号を、所定のビットに対応するデータメモリセル23-iに供給して、そのセル23-iのトランジスタTr11をオン状態にして、コンデンサC11の端子電圧を、その演算結果に対応する電圧にする。

【0204】なお、ALUセル24Aにおける演算動作は、プログラム制御部27Aより供給されるALU制御信号（SALU-CONT）に従って行われる。ALUセル24Aにおける演算結果は、上述のようにデータメモリ部23に書き込まれるか、あるいは、必要に応じてALUセル24Aのフリップフロップ82-3に記憶される。また、ALU81における演算が加算の場合は、ALUセル24Aは、演算結果におけるキャリーをフリップフロップ82-3に、サムをデータメモリ部23に出力する。

【0205】次に、データメモリセル23-iよりデータを出力させる場合、プログラム制御部27Aは、出力するデータを記憶しているデータメモリセル23-iに、メモリアクセス信号（SRAAあるいはSRBA）を供給して、そのセル23-iのトランジスタTr13またはTr14をオン状態にして、コンデンサC11に記憶されているデータを読み出しビット線61または62に出力する。

【0206】そして、プログラム制御部27Aは、ALUセル24Aに、所定の制御信号を供給し、データメモリセル23-iからのデータを、出力SAMセル25-iに転送させる。このとき、プログラム制御部27Aは、その出力SAMセル25-iのコンデンサC4にデータが供給されるように、SOW信号を出力し、そのセルのトランジスタTr17をオン状態にして、コンデンサC4の端子電圧を、そのデータに応じた電圧にする。

【0207】なお、データは、書き込みビット線63を経由して、1ビットずつ転送される。このとき、データに対して、ALU81により何らかの処理を行ってもよい。

【0208】また、出力SAM部25の各出力SAMセル25-iにデータを記憶させるときに利用するSOW

信号と、データメモリ部23の各データメモリセル23-iからデータを読み出すときに利用するメモリアクセス信号(SRAA, SRBA)は、同じアドレス空間内のアドレスであり、それぞれロウデコーダでデコードされてワード線を介して供給される。

【0209】出力ポインタ26が指定した要素プロセッサ31の出力SAMセル25-iにおいては、トランジスタTr8が出力ポインタ信号に対応してオン状態になり、コンデンサC4の電位に応じた出力信号が出力データバス66に出力される。

【0210】そして、出力ポインタ26は、値が「H」である出力ポインタ信号を、左端の要素プロセッサ31から右端の要素プロセッサ31まで、クロック信号に従って、順次供給することにより、各要素プロセッサ31の出力SAMセル25-iから、データを順次出力させる。

【0211】このようにして、出力SAMセル25-iに供給されたデータは、出力データバス66を介して出力端子DOOUTに出力される。

【0212】なお、要素プロセッサ31は、画像信号の1水平走査期間の画素数H以上設けられているので、この動作を行うことにより、出力画像信号の1水平走査期間分のデータが、出力SAM部25より出力される。そして、この出力動作は水平走査期間毎に繰り返される。

【0213】以上のようにして、各要素プロセッサ31は、データの入力、データの転送、演算、データの出力などの処理を行う。

【0214】なお、第2の実施の形態においては、起動時や、水平帰線期間または垂直帰線期間に、すべてのフィルタ係数セットが、すべての要素プロセッサ31のデータメモリ部23に供給される。このとき、フィルタ係数セットは、メモリ28から、入力データバス65の一部(所定のビット幅)を介して、入力SAM部22に供給され、データメモリ部23に転送される。このときの動作は、次に説明する、データメモリ部23への入力データRiの供給の動作と同様であるので、その説明を省略する。

【0215】次に、図7のフローチャートを参照して、第2の実施の形態の動作について説明する。

【0216】最初に、ステップS1において、1水平走査期間分の所定のLビットの入力データRi(= {ri0, ..., ri(L-1)})が入力SAM部22に入力される。

【0217】(10/7)倍の画像の拡大を行う場合、上述のように、各出力データの算出に必要な入力データを保持している要素プロセッサ31と、その出力データを算出する要素プロセッサ31との位置関係が、出力データ毎に変化する。例えば、7画素の入力データに対応して10画素の出力データを算出する場合、各出力データの算出に必要な入力データを保持している要素プロセ

ッサ31のパターンは、図35に示すように5つのパターンに分類される。

【0218】そこで、このとき、図8に示すように、7個の入力データのいずれかを重複させて、その7個の入力データを、10個の要素プロセッサ31に密に供給する。即ち、図34に示す要素プロセッサ31のうち、入力データが供給されないものに対して、その左隣りの要素プロセッサ31と同一の入力データを供給する。

【0219】この入力データの供給の手順は、図30の装置におけるデータの入力と同様にして実現してもよいし、あるいは、特開平8-123683号公報に記載されている方法を利用して、疎な状態になるようにデータを一旦供給し、その後、プログラムに従って所定のデータをコピーするようにして実現してもよい。

【0220】なお、図8においては、入力データRiおよび出力データQiは、実際は、8ビット程度であるが、便宜上、それぞれ4ビットで表されている。また、入力SAM部22、データメモリ部23、および、出力SAM部25においては、説明に必要なメモリ容量だけを示している。

【0221】このように入力データを供給することにより、図9に示すように、例えば、図9の種類1のパターンの場合、左2つ隣りの要素プロセッサ31と、左隣りの要素プロセッサ31に、同一の入力データが供給されるとともに、右隣りの要素プロセッサ31と、右2つ隣りの要素プロセッサ31に、同一の入力データが供給されるので、図9の種類1のパターンは、種類2のパターンと同様に取り扱うことができる。

【0222】また、図9の種類3のパターンの場合、所定の要素プロセッサ31と、その左隣りの要素プロセッサ31に、同一の入力データが供給されるので、図9の種類3のパターンは、種類4のパターンと同様に取り扱うことができる。

【0223】さらに、また、図9の種類5のパターンの場合、右隣りの要素プロセッサ31と、右2つ隣りの要素プロセッサ31に、同一の入力データが供給されるので、図9の種類5のパターンは、種類2のパターンと同様に取り扱うことができる。

【0224】従って、図8に示すように入力データを供給することにより、上述の5つのパターンは、図10に示す2つパターン(種類2と種類4)に縮退している。

【0225】なお、(10/7)以外の変換比率の場合も、予め、上述のパターンの数が最小になるような入力データの供給方式を算出しておくことにより、上述のように位置関係のパターンを縮退させることができる。

【0226】そして、プログラム制御部27Aは、2つのパターンについて、要素プロセッサ31に対応するパターンを示す1ビットの値(0または1)を、入力データとともに、各要素プロセッサ31に供給する。

【0227】次に、ステップS2乃至ステップS5にお

いて、プログラム制御部27Aは、各要素プロセッサ31に供給された入力データ R_i を、入力SAM部22から、書き込みビット線63を介して、データメモリ部23に1ビットずつ転送させる。

【0228】今の場合、入力データ R_i は、便宜上、4ビットに設定されており、入力SAM部22のアドレス0乃至4に記憶されている。従って、図8に示すように、入力SAM部22のアドレス0の内容が、データメモリ部23のアドレス8に転送され、同様に、入力SAM部22のアドレス1乃至3の内容が、データメモリ部23のアドレス9乃至11にそれぞれ転送される。

【0229】そして、ステップS6において、各要素プロセッサ31は、後述の信号処理（補間演算）を行う。

【0230】ステップS7乃至ステップS10において、プログラム制御部27Aは、各要素プロセッサ31において算出された演算結果（出力データ Q_i ）を、データメモリ部23から、読み出しビット線61、62、および、ALUセル24Aを介して出力SAM部25に1ビットずつ転送させる。

【0231】今の場合、出力データ Q_i （= q_{i0}, \dots, q_{i3} ）は、便宜上、4ビットに設定されており、データメモリ部23のアドレス16乃至19に記憶されている。従って、図8に示すように、データメモリ部23のアドレス16の内容が、出力SAM部25のアドレス20に転送され、同様に、データメモリ部23のアドレス17乃至19の内容が、出力SAM部25のアドレス21乃至23にそれぞれ転送される。

【0232】次に、ステップS11において、算出された1水平走査期間分の出力データ Q_i が出力SAM部25から出力される。

【0233】以上のようにして、各1水平走査期間分の画像データ毎に、フィルタ演算が行われる。なお、ステップS1の動作、ステップS2乃至ステップS10の動作、および、ステップS11の動作は並列に行われており、所定の1水平走査期間分の画像データに対してステップS2乃至ステップS10の動作が行われているとき、1ライン前の1水平走査期間分の画像データに対してステップS11の動作が行われ、1ライン後の1水平走査期間分画像データに対してステップS1の動作が行われる。

【0234】次に、図11および図12のフローチャートを参照して、図7のステップS6における信号処理の詳細について説明する。

【0235】最初にステップS21において、各要素プロセッサ31は、供給されたデータを保持するとともに、左隣りの要素プロセッサ31にコピーする。なお、本実施の形態においては、要素プロセッサ31は、左右1つ隣り、または、左右2つ隣りの要素プロセッサだけと通信を行うことが可能であるように構成されており、補間演算を行う場合、右3つ隣りの要素プロセッサ31

に供給されたデータが使用されるので、ステップS21において、そのデータが、左隣りの要素プロセッサ31に予めコピーされる。

【0236】なお、以下、所定の要素プロセッサ31に供給されたデータを R_0 とし、左隣りの要素プロセッサ31に供給されたデータを R_{-1} とし、左2つ隣りの要素プロセッサ31に供給されたデータを R_{-2} とする。また、右隣りの要素プロセッサ31に供給されたデータを R_{+1} とし、右2つ隣りの要素プロセッサ31に供給されたデータを R_{+2} とし、右3つ隣りの要素プロセッサ31に供給されたデータを R_{+3} とする。

【0237】次にステップS22において、要素プロセッサ31は、左隣りの要素プロセッサ31のデータ R_{-1} と、予め供給されているフィルタ係数 FC_1 の積を演算し、その演算結果を Y_{1A} に代入する。この積の演算は、所定の回数のビット演算を行うことにより実行される。

【0238】なお、補間演算に利用されるフィルタ係数セット（ FC_1, FC_2, FC_3, FC_4 ）には、そのフィルタ係数セットに対応する位相が、その要素プロセッサによって処理される画素の位相に最も近いものが選択される。

【0239】このように、所定の数のフィルタ係数セットを用意しておき、処理される画素の位相に対応する最適なフィルタ係数セットを使用することにより、変換比率が変更されても、同様の動作で、最適なフィルタ係数セットが選択されることになる。

【0240】ステップS23において、要素プロセッサ31は、自分に供給されたデータ R_0 と、フィルタ係数 FC_2 の積を演算し、その演算結果を Y_{2A} に代入する。

【0241】そして、ステップS24において、要素プロセッサ31は、 Y_{1A} と Y_{2A} の和を演算し、その演算結果を Y_{1A} に代入する。この和の演算は、所定の回数のビット演算を行うことにより実行される。

【0242】次にステップS25において、要素プロセッサ31は、右2つ隣りの要素プロセッサ31のデータ R_{+2} と、フィルタ係数 FC_3 の積を演算し、その演算結果を Y_{2A} に代入する。

【0243】そして、ステップS26において、要素プロセッサ31は、 Y_{1A} と Y_{2A} の和を演算し、その演算結果を Y_{1A} に代入する。

【0244】次にステップS27において、要素プロセッサ31は、右2つ隣りの要素プロセッサ31が有している、右3つ隣りの要素プロセッサ31のデータ R_{+3} と、フィルタ係数 FC_4 の積を演算し、その演算結果を Y_{2A} に代入する。

【0245】そして、ステップS28において、要素プロセッサ31は、 Y_{1A} と Y_{2A} の和を演算し、その演算結果を Y_{1A} に代入する。このとき、 Y_{1A} の値は、 $R_{-1} \times FC_1 + R_0 \times FC_2 + R_{+2} \times FC_3 + R_{+3} \times FC_4$ であり、図10の種類2のパターンに対応している。

【0246】次に、ステップS29において、要素プロセッサ31は、左2つ隣りの要素プロセッサ31のデータR-2と、フィルタ係数FC1の積を演算し、その演算結果をY1Bに代入する。

【0247】ステップS30において、要素プロセッサ31は、自分に供給されたデータR0と、フィルタ係数FC2の積を演算し、その演算結果をY2Bに代入する。

【0248】そして、ステップS31において、要素プロセッサ31は、Y1BとY2Bの和を演算し、その演算結果をY1Bに代入する。

【0249】次にステップS32において、要素プロセッサ31は、右隣りの要素プロセッサ31のデータR+1と、フィルタ係数FC3の積を演算し、その演算結果を、Y2Bに代入する。

【0250】そして、ステップS33において、要素プロセッサ31は、Y1BとY2Bの和を演算し、その演算結果をY1Bに代入する。

【0251】次にステップS34において、要素プロセッサ31は、右2つ隣りの要素プロセッサ31のデータR+2と、フィルタ係数FC4の積を演算し、その演算結果をY2Bに代入する。

【0252】ステップS35において、要素プロセッサ31は、Y1BとY2Bの和を演算し、その演算結果をY1Bに代入する。このとき、Y1Bの値は、 $R-2 \times FC1 + R0 \times FC2 + R+1 \times FC3 + R+2 \times FC4$ であり、図10の種類4のパターンに対応している。

【0253】そして、ステップS36において、要素プロセッサ31は、入力データRiとともに供給された、上述の位置関係を示す値(0または1)を参照して、その値が、第1の値(図10の種類2に対応する値)であるか否かを判断し、その値が第1の値であると判断した場合、ステップS37に進み、ステップS28のY1Aを演算結果とし、上述の位置関係を示す値が、第1の値ではないと判断した場合(即ち、図10の種類4に対応する値である場合)、ステップS35のY1Bを演算結果とする。

【0254】以上のようにして、2種類の位置関係に対応して近傍の要素プロセッサ31のデータを利用してフィルタ演算(補間演算)が行われる。そして、上述のように位相の誤差が最小になるようにフィルタ係数セットを選択することにより、SIMD形式の並列プロセッサにおいても、任意の変換比率の画像処理を行うことができる。

【0255】なお、セレクトを設けて、メモリ28からのフィルタ係数セットおよび入力データのいずれか一方を、入力SAM部22に供給させるようにすることにより、例えば、垂直帰線期間などの入力SAM部22が入力データRiの供給に利用されていない期間に、フィルタ係数セットを入力データと同様に供給するようにしてもよい。

【0256】このようにすることにより、入力データと同一のビット数のバス65を利用して、フィルタ係数の供給を行うことができるので、ビット数の大きい(語長の長い)フィルタ係数を短時間で供給することができる。

【0257】例えばフィルタ係数のビット数が10である場合、4つのフィルタ係数のセットは合計40ビットのデータであるので、例えば16ビットの入力データバス65および入力SAM部22を介してデータメモリ部23に、垂直帰線期間内で、フィルタ係数を供給することが十分可能である。

【0258】なお、第2の実施の形態においては、フィルタ係数セットは、入力データRiとは異なるパターンで入力SAM部22に供給されるので(フィルタ係数は、その順番通りに、各要素プロセッサ31に供給されるので)、入力データと並行してフィルタ係数を供給する場合、2系統の入力SAM部22のポインタ制御用の回路を設け、入力データRiに対するポインタ制御と、フィルタ係数に対するポインタ制御が、独立に行われるようにする。

【0259】また、この入力データの供給の手順は、図30の装置におけるデータの入力と同様にして実現してもよいし、あるいは、特開平8-123683号公報に記載されている方法を利用して、疎な状態になるようにデータを一旦供給し、その後、プログラムに従って所定のデータをコピーするようにして実現してもよい。

【0260】図13は、本発明の第3の実施の形態の構成を示している。第3の実施の形態においては、メモリ29(記憶手段)は、図4のメモリ28と同様に、所定の分割数に対応するフィルタ係数セットのデータを保持するようになされている。また、メモリ29は、プログラム制御部27Bにより制御され、起動時、水平帰線期間、垂直帰線期間などにおいて、フィルタ係数セットを、そのフィルタ係数セットに対応する位相の画素の値を算出する要素プロセッサ31のALUアレイ部24を介してデータメモリ部23に供給する。

【0261】メモリ28Aは、各要素プロセッサ31に対応して、その要素プロセッサ31により算出される画素の位相に対応するフィルタ選択番号i(即ち、位相xが、補間する画素の位相に最も近いフィルタ係数セットのフィルタ選択番号Piに対応する番号i)を保持する。そのフィルタ選択番号iは、第2の実施の形態のフィルタ係数セットと同様に、入力データバス65を介して、入力データRiとともにデータメモリ部23に供給される。

【0262】なお、メモリ28A、29に記憶されているデータは、外部の制御用CPUにより、起動時などに予め供給されるものとする。

【0263】プログラム制御部27Bは、各部を制御し、後述の動作を行わせるようになされている。

【0264】なお、その他の構成要素は、第2の実施の形態のものと同様であるので、その説明を省略する。

【0265】なお、メモリ28Aに保持されているフィルタ選択番号*i*は、例えば起動時に、入力データバス65および入力SAM部22を介してデータメモリ部23に予め供給されているものとする。

【0266】例えば、画素間隔の分割数（即ち、フィルタ係数セットの数）が16である場合、メモリ28Aは、水平方向の画素数*H*に関係なく、16種類の位相に対応する16個のフィルタ選択番号*i*を記憶していればよい。即ち、フィルタ選択番号*i*が16個存在する場合、そのフィルタ選択番号は、4ビットの2進数で表現することができるので、メモリ28Aは、フィルタ選択番号*i*として4ビットのデータを記憶する。

【0267】また、フィルタ選択番号*i*が1,000種類あったとしても、10ビットの2進数で表現することができるので、第2の実施の形態のように、フィルタ係数を入力SAM部22を介して供給する場合より、メモリ28Aの容量を低減することができる。

【0268】図14は、各要素プロセッサ31のデータメモリ部23に記憶されているフィルタ選択番号*i*（＝ $\{\phi_{i0}, \dots, \phi_{i3}\}$ ）の一例を示している。図14のデータメモリ部23においては、16種類のフィルタ選択番号のうち、10種類のフィルタ選択番号*i*（*i*＝0, ..., 9）が、4ビットのデータとして記憶されている。例えば、番号が6である要素プロセッサ31のデータメモリ部23には、フィルタ選択番号*i*が3である4ビットのデータ $\{\phi_{30}, \dots, \phi_{33}\}$ が記憶されている。

【0269】次に、図15のフローチャートを参照して、第3の実施の形態において、フィルタ係数セットを各要素プロセッサ31のデータメモリ部23に供給するときの各部の動作について説明する。

【0270】最初にステップS41において、プログラム制御部27Bは、供給するフィルタ係数セットに対応するフィルタ選択番号*i*をカウントするカウンタ*j*の値を0に設定する。

【0271】次に、ステップS42において、プログラム制御部27Bは、カウンタ*j*の値をビット単位で供給するときに利用するカウンタ*m*の値を1に設定する。

【0272】そして、プログラム制御部27Bは、すべての要素プロセッサ31のALUセル24Aに、カウンタ*j*の値の第*m*番目のビットの値を出力し、各要素プロセッサ31のALUセル24Aは、そのビットの値を受け取る。

【0273】ステップS44において、プログラム制御部27Bは、カウンタ*m*の値がカウンタ*j*のビット長以上であるか否かを判断し、カウンタ*m*の値がカウンタ*j*のビット長より小さいと判断した場合、ステップS45において、カウンタ*m*の値を1だけ増加させた後、ステ

ップS43に戻り、次のビットの供給を行う。

【0274】このようにして、カウンタ*j*の値は、1ビットずつ各要素プロセッサ31に供給される。

【0275】一方、ステップS44においてカウンタ*m*の値がカウンタ*j*のビット長以上であると判断された場合、カウンタ*j*の値が供給されたことになるので、ステップS46において、各要素プロセッサ31は、受信したカウンタ*j*の値と、予めメモリ28Aより供給されたフィルタ選択番号*i*の値が同一であるか否かを判断し、同一である場合、例えば、その判断に対応してフラグを設定し、ステップS47に進む。

【0276】ステップS47において、各要素プロセッサ31は、そのフラグに対応して、供給されたフィルタ係数セットのビット数をカウントするカウンタ*k*の値を1に設定する。

【0277】そして、ステップS48において、各要素プロセッサ31は、メモリ29より出力された、フィルタ係数セットの第*k*番目のビットの値をALUセル24Aで受け取り、データメモリ部23に記憶させる。

【0278】なお、メモリ29においては、各位相（即ち、フィルタ選択番号*i*）に対応するフィルタ係数セットが、係数毎に、最上位ビット（MSB）または最下位ビット（LSB）から順番に記憶されており、フィルタ係数セットは、上述のように1ビットの回線を介して、1ビットずつ要素プロセッサ31のALUセル24Aに順次出力される。

【0279】そして、ステップS49において、各要素プロセッサ31は、カウンタ*k*の値がフィルタ係数セットのビット長以上であるか否かを判断し、カウンタ*k*の値がフィルタ係数セットのビット長より小さいと判断した場合、ステップS50において、カウンタ*k*の値を1だけ増加させた後、ステップS48に戻り、フィルタ係数セットの次のビットの受信を行う。

【0280】一方、ステップS49においてカウンタ*k*の値がフィルタ係数セットのビット長以上であると判断された場合、カウンタ*j*の値に対応するフィルタ係数セットの供給が終了したことになるので、ステップS51に進む。

【0281】一方、ステップS46において、要素プロセッサ31が、カウンタ*j*の値と、予めメモリ28Aより供給されたフィルタ選択番号*i*の値が同一ではないと判断した場合（即ち、フラグが立っていない場合）、その要素プロセッサ31は、メモリ29より出力されたフィルタ係数セットを受け取らず、即ち、データメモリ部23に記憶せず、ステップS47乃至S50をスキップする。なお、実際には、SIMD制御において処理のスキップを行うことは困難であるので、要素プロセッサ31は、ステップS47乃至S50をスキップした場合と同一の結果になるように処理を行う。

【0282】次に、ステップS51において、プログラ

ム制御部27Bは、カウンタjの値が、フィルタ係数セットの数Nより1だけ減算した値以上であるか否かを判断し、カウンタjの値が、フィルタ係数セットの数Nより1だけ減算した値以上($j \geq N-1$)であると判断した場合、N個のフィルタ係数セットのうちのいずれかが各要素プロセッサ31に供給されたことになるので、フィルタ係数セットの供給の処理を終了する。

【0283】一方、プログラム制御部27Bは、カウンタjの値が、フィルタ係数セットの数Nより1だけ減算した値より小さい($j < N-1$)と判断した場合、ステップS52において、カウンタjの値を1だけ増加させて、ステップS42に戻り、次のフィルタ選択番号iに対応するフィルタ係数セットの供給を行う。

【0284】このようにして、各要素プロセッサ31は、予め供給されているフィルタ選択番号iに対応するフィルタ係数をメモリ29より受け取り、データメモリ部23に記憶させる。

【0285】このように、フィルタ係数セットを入力データRiとは別の経路で供給することにより、フィルタ係数セットを選択的に要素プロセッサ31に供給することを、多くのプログラムステップを必要とすることなく、容易に実現することができる。

【0286】メモリ29に記憶されている例えば16種類のフィルタ係数セットのうちのいずれかを各要素プロセッサ31に供給する場合、1つのフィルタ係数セットが、すべての要素プロセッサ31の約16分の1の要素プロセッサ31に同時に供給されるので、フィルタ係数セットが例えば40ビットであるとき、要素プロセッサ31の数に拘わらず、640($=40 \times 16$)ステップの動作で、すべての要素プロセッサ31にフィルタ係数セットを供給することができる。

【0287】なお、画像データの処理を行うときの動作は、第2の実施の形態のものと同様であるので、その説明を省略する。

【0288】以上のように、第3の実施の形態においては、フィルタ係数セットを、入力データとは別の経路で供給するので、入力SAM部22の稼働状況に拘わらずフィルタ係数セットを供給することができる。

【0289】なお、第3の実施の形態においては、変換比率を変更させる場合、メモリ28Aに記憶されている、各要素プロセッサ31に対応するフィルタ選択番号iを変更すればよい。

【0290】次に、本発明の第4の実施の形態について説明する。第4の実施の形態においては、各要素プロセッサ31が、フィルタ選択番号iに対応して、各要素プロセッサ31で、フィルタ係数セットを算出するようになされている。

【0291】なお、第4の実施の形態の構成およびフィルタ演算時の動作は、第2の実施の形態と同様であるので、その説明を省略する。ただし、メモリ28は、第3

の実施の形態のメモリ28Aと同様にフィルタ選択番号iを記憶しているものとする。

【0292】従って、変換比率を変更させる場合、メモリ28に記憶されている、各要素プロセッサ31に対応するフィルタ選択番号iを変更すればよい。

【0293】次に、図16および図17のフローチャートを参照して、第4の実施の形態において、フィルタ係数セットを算出するときの各部の動作について説明する。なお、ここでは、式(4)のCubic近似法におけるフィルタ係数セットを算出する。勿論、他の近似法におけるフィルタ係数セットを算出するようにしてもよい。

【0294】最初に、ステップS61において、要素プロセッサ31は、画像の変換比率をK/L倍とすると、予め供給されているフィルタ選択番号iとKより、値を算出する画素の位相 i/K を算出し、 X_0 として記憶する。なお、KおよびLは、プログラム制御部27Aより供給される。

【0295】次に、要素プロセッサ31は、ステップS62において、 X_0 をXに代入し、ステップS63において、Xの2乗($X \times X$)を演算し、その演算結果を X_2 として記憶する。

【0296】さらに、ステップS64において、要素プロセッサ31は、 X_2 とXの積(即ち、Xの3乗)を演算し、その演算結果を X_3 として記憶する。

【0297】そして、ステップS65において、要素プロセッサ31は、式(4)を利用して、X、 X_2 および X_3 より、次式に従ってフィルタ係数FC3を算出する。

$$FC3 = -X_3 + 5X_2 - 8X + 4 \quad (5)$$

【0298】次に、ステップS66において、要素プロセッサ31は、 $X_0 (= i/K)$ に1だけ加算した値を、Xに代入する。

【0299】そして、要素プロセッサ31は、ステップS67において、Xの2乗($X \times X$)を演算し、その演算結果を X_2 に代入し、ステップS68において、 X_2 とXの積(即ち、Xの3乗)を演算し、その演算結果を X_3 に代入する。

【0300】ステップS69において、要素プロセッサ31は、式(4)を利用して、X、 X_2 および X_3 より、次式に従ってフィルタ係数FC4を算出する。

$$FC4 = X_3 - 2X_2 + 1 \quad (6)$$

【0301】次に、ステップS70において、要素プロセッサ31は、1から X_0 を減算した値をXに代入する。

【0302】そして、要素プロセッサ31は、ステップS71において、Xの2乗($X \times X$)を演算し、その演算結果を X_2 に代入し、ステップS72において、 X_2 とXの積(即ち、Xの3乗)を演算し、その演算結果を X_3 に代入する。

【0303】ステップS73において、要素プロセッサ31は、式(4)を利用して、 X 、 X_2 および X_3 より、次式に従ってフィルタ係数 $FC2$ を算出する。

$$FC2 = -X_3 + 5X_2 - 8X + 4 \quad (7)$$

【0304】次に、ステップS74において、要素プロセッサ31は、 X に1だけ加算した値 $(2-i/K)$ を演算し、その演算結果を X に代入する。

【0305】そして、要素プロセッサ31は、ステップS75において、 X の2乗($X \times X$)を演算し、その演算結果を X_2 に代入し、ステップS76において、 X_2 と X の積(即ち、 X の3乗)を演算し、その演算結果を X_3 に代入する。

【0306】ステップS77において、要素プロセッサ31は、式(4)を利用して、 X 、 X_2 および X_3 より、次式に従ってフィルタ係数 $FC1$ を算出する。

$$FC1 = X_3 - 2X_2 + 1 \quad (8)$$

【0307】以上のようにして、第4の実施の形態においては、フィルタ選択番号 i に対応して、各要素プロセッサ31で、フィルタ係数セット($FC1$ 、 $FC2$ 、 $FC3$ 、 $FC4$)を算出する。

【0308】このように、フィルタ係数セットを各要素プロセッサ31で算出することにより、要素プロセッサ31の外部のメモリ(メモリ28、29など)よりフィルタ係数セットを供給する必要がなく、フィルタ係数セットの供給のタイミングなどを考慮する必要がなくなる。

【0309】図18は、本発明の第5の実施の形態の構成を示している。

【0310】第5の実施の形態は、第3の実施の形態のメモリ28Aを取り除いたもので、フィルタ選択番号 i を、各要素プロセッサ31で算出するようになされている。

【0311】なお、プログラム制御部27Cのプログラムは、後述の動作を行わせるように変更されているが、その他の構成要素、フィルタ係数の供給時の動作、および、フィルタ演算時の動作は、第3の実施の形態のものと同様であるので、その説明を省略する。

【0312】次に、図19のフローチャートを参照して、フィルタ選択番号 i を演算するときの動作について説明する。

【0313】最初にステップS81において、要素プロセッサ31は、作業空間としてレジスタ ZA_0 、 ZB_0 、 ZC_0 をそれぞれ確保する。

【0314】次に、ステップS82において、各要素プロセッサ31は、 ZA_0 、 ZB_0 、および、 ZC_0 に零をそれぞれ代入する。

【0315】ステップS83において、各要素プロセッサ31は、左隣の要素プロセッサ31の ZA_0 の値 ZA_{-1} と、変換比率を K/L 倍(即ち、 $K:L$)としたときの L との和を演算し、その演算結果を、 ZA_0 に記憶

する。なお、最も左側の要素プロセッサ31は、左隣りに要素プロセッサ31がないので、 ZA_{-1} を0として演算を行う。

【0316】ステップS84において、各要素プロセッサ31は、 ZA_0 の値が K より大きいかなかを判断し、 ZA_0 の値が K より大きいと判断した場合、ステップS85において、 ZA_0 の値を K で割算したときの剰余を演算し、その演算結果を ZA_0 に代入する。

【0317】一方、各要素プロセッサ31は、 ZA_0 の値が K 以下であると判断した場合、ステップS85をスキップする。なお、実際には、SIMD制御において処理のスキップを行うことは困難であるので、要素プロセッサ31は、ステップS85をスキップした場合と同一の結果になるように処理を行う。

【0318】そして、ステップS86において、各要素プロセッサ31は、ステップS83乃至ステップS85の動作を、現在取り扱っている画像フォーマットの水平方向の画素数より多く繰り返したか否かを判断し、ステップS83乃至ステップS85の動作を、現在取り扱っている画像フォーマットの水平方向の画素数以下しか繰り返していないと判断した場合、ステップS83に戻り、ステップS83乃至ステップS85の動作を再度行う。

【0319】一方、各要素プロセッサ31は、ステップS83乃至ステップS85の動作を、現在取り扱っている画像フォーマットの水平方向の画素数より多く繰り返したと判断した場合、ステップS87に進む。

【0320】ステップS87において、各要素プロセッサ31は、左隣の要素プロセッサ31の ZB_0 の値 ZB_{-1} と L との和を演算し、その演算結果を ZC_0 に記憶する。なお、最も左側の要素プロセッサ31は、左隣りに要素プロセッサ31がないので、 ZB_{-1} を0として演算を行う。

【0321】次に、ステップS88において、要素プロセッサ31は、 ZC_0 の値が K の2倍の値より大きいかなかを判断し、 ZC_0 の値が K の2倍の値より大きいと判断した場合、ステップS90において、 ZB_0 の値から K を減算した値を ZB_0 に代入する。

【0322】一方、要素プロセッサ31は、 ZC_0 の値が K の2倍の値以下であると判断した場合、ステップS89において、 ZC_0 の値から K を減算した値を ZB_0 に代入する。

【0323】そして、ステップS91において、各要素プロセッサ31は、ステップS87乃至ステップS90の動作を、現在取り扱っている画像フォーマットの水平方向の画素数より多く繰り返したか否かを判断し、ステップS87乃至ステップS90の動作を、現在取り扱っている画像フォーマットの水平方向の画素数以下しか繰り返していないと判断した場合、ステップS87に戻り、ステップS87乃至ステップS90の動作を再度行

う。

【0324】一方、各要素プロセッサ31は、ステップS87乃至ステップS90の動作を、現在取り扱っている画像フォーマットの水平方向の画素数より多く繰り返したと判断した場合、ステップS92に進む。

【0325】そして、ステップS92において、各要素プロセッサ31は、KがLより大きい、即ち、画像の拡大の処理であるか否かを判断し、KがLより大きいと判断した場合、ステップS94において、フィルタ選択番号iとしてZ A₀の値を利用し、KがL以下であると判

断した場合、ステップS93において、フィルタ選択番号iとしてZ B₀の値を利用する。

【0326】以上のようにして、フィルタ選択番号iを算出する。なお、ステップS85において、割算(剰余の算出)を行っているが、実際には、減算を繰り返し実行している。なお、上述の処理は多くの処理ステップを有するが、リアルタイムの処理を行う前や、垂直帰線期間などにおいて処理を行うことにより特に問題は生じない。

【0327】なお、ステップS84およびステップS88における判断に対応して、入力データまたは出力データと、要素プロセッサ31との対応関係(図34のR_iの入力の仕方)を設定するようにしてもよい。即ち、ステップS85は、上述の位相のモジュロ演算と同様の処理を行っているので、ステップS84における判断に対応して、モジュロ演算が発生した要素プロセッサ31が、図34の入力データがない場所であると判断される。

【0328】図20は、本発明の第6の実施の形態の構成を示している。

【0329】第6の実施の形態は、フィルタ選択番号iおよびそれに対応するフィルタ係数セットを、第4の実施の形態または第5の実施の形態と同様に、各要素プロセッサ31で算出することにより、メモリ28、28A、29を不要としたものである。

【0330】プログラム制御部27Dは、各要素プロセッサ31を制御し、第4の実施の形態または第5の実施の形態のプログラム制御部と同様に、フィルタ選択番号iおよびそれに対応するフィルタ係数セットを算出させるようになされている。

【0331】その他の構成要素は、第5の実施の形態と同様であるので、その説明を省略する。また、フィルタ係数セットを算出するときの動作は、第4の実施の形態と同様であり、その他の動作は、第5の実施の形態と同様であるので、その説明を省略する。

【0332】なお、上記実施の形態において、位相の誤差を低減するために分割数を大きくする場合、フィルタ演算を2段階にすることにより、処理を簡単にすることができる。

【0333】例えば、分割数が16であるCubic近

似を分割数が256であるCubic近似に変更する代わりに、最初に、分割数が16のCubic近似を行い、その演算結果である補間値のうちの近傍の2点から、分割数が16の例えば双1次近似を行うことにより、全体の補間の処理が簡単になる。

【0334】また、画素データは、通常、輝度情報を含む輝度データと、色情報を含む色データで構成されているので、精細度が要求される輝度データに対しては、色データより大きい分割数で補間の処理を行うようにしてもよい。

【0335】なお、上記実施の形態においては、主に画像の拡大について説明しているが、画像の縮小を行うことも勿論可能である。また、画像の縮小の場合、入力SAM部22には、入力データが順番通りに密に供給され、出力SAM部25には、出力データが疎に出力されてくる。

【0336】また、上記実施の形態の端部(右端、左端)周辺の要素プロセッサ31においては、演算に利用する入力データを有する周辺の要素プロセッサ31が存在しない場合があるので、その場合においては、その入力データの値を0として演算を行う。

【0337】その他に、例えば、端のデータがその外側に連続している、あるいは、端を中心にしてデータが対称になっていると仮定するなど、画像端辺での処理には様々な方法が考えられる。これらの方法のうち、所定の方法に対応してプログラムを記述することにより、その方法を実現することができる。

【0338】なお、上述のメモリ28、28A、29の容量は、画素の位相の種類や、変換比率に対応する分数の分子または分母に依存する量であり、それほど大きくないので、装置の規模が大きくなるようなことはない。

【0339】

【発明の効果】以上のごとく、請求項1に記載の画像処理装置および請求項3に記載の画像処理方法によれば、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットを記憶している記憶部から、位相が、補間演算される画素データの位相に最も近いフィルタ係数セットを演算部に出力し、演算部により、フィルタ係数セットを利用して、画素データの補間演算を行うようにしたので、任意の変換比率の画像の拡大または縮小を行うことができる。

【0340】請求項4に記載の画像処理装置および請求項19に記載の画像処理方法によれば、所定の分割数で元の画像の画素間隔を分割したときの各位相に対応するフィルタ係数セットのうち、位相が、処理される画素データの位相に最も近いフィルタ係数セットを、要素プロセッサにそれぞれ供給し、要素プロセッサが、フィルタ係数セットを利用して、画素データの補間の処理をそれぞれ行うようにしたので、SIMD形式の並列プロセッサを利用して任意の変換比率の画像の拡大または縮小を

行うことができる。

【図面の簡単な説明】

【図1】本発明の画像処理装置の第1の実施の形態の構成を示すブロック図である。

【図2】フィルタ選択信号 P_i および位相量 x と、フィルタ係数セットとの対応関係の一例を示す図である。

【図3】図1の装置において行われるフィルタ演算の各サイクルにおける補間値 Q_i と、フィルタ選択信号 P_i の対応関係の一例を示す図である。

【図4】本発明の画像処理装置の第2の実施の形態の構成を示すブロック図である。

【図5】要素プロセッサの構成例を示すブロック図である。

【図6】要素プロセッサの詳細な構成例を示す回路図である。

【図7】図4の画像処理装置の動作を説明するフローチャートである。

【図8】図4の画像処理装置の各部に記憶されるデータの一例を示す図である。

【図9】処理に必要なデータを有する要素プロセッサとの位置関係の例を示す図である。

【図10】図9の位置関係を縮退させた位置関係の例を示す図である。

【図11】図4の画像処理装置におけるフィルタ演算の処理を説明するフローチャートである。

【図12】図4の画像処理装置におけるフィルタ演算の処理を説明するフローチャートである。

【図13】本発明の画像処理装置の第3の実施の形態の構成を示すブロック図である。

【図14】データメモリ部に記憶されているフィルタ選択番号の一例を示す図である。

【図15】フィルタ係数セットを供給するときの図13の画像処理装置の動作について説明するフローチャートである。

【図16】第4の実施の形態において、各要素プロセッサがフィルタ係数セットを演算するときの動作について説明するフローチャートである。

【図17】第4の実施の形態において、各要素プロセッサがフィルタ係数セットを演算するときの動作について説明するフローチャートである。

【図18】本発明の画像処理装置の第5の実施の形態の構成を示すブロック図である。

【図19】各要素プロセッサがフィルタ選択番号を演算するときの図18の画像処理装置の動作について説明するフローチャートである。

【図20】本発明の画像処理装置の第6の実施の形態の構成を示すブロック図である。

【図21】原画像の一例を示す図である。

【図22】原画像を拡大した画像の一例を示す図である。

【図23】原画像の画素と、拡大した画像の画素の位置関係の一例を示す図である。

【図24】原画像の解像度を高くした画像の一例を示す図である。

【図25】原画像を縮小した画像の一例を示す図である。

【図26】原画像の画素と、縮小した画像の画素の位置関係の一例を示す図である。

【図27】原画像の解像度を低くした画像の一例を示す図である。

【図28】原画像の画素と、補間により生成される画素の位置関係の一例を示す図である。

【図29】補間関数の例を示す図である。

【図30】ハードウェア的にフィルタ演算を行う装置の一構成例を示すブロック図である。

【図31】図30の装置において行われるフィルタ演算の各サイクルにおける各部の信号の一例を示す図である。

【図32】フィルタ選択信号とフィルタ係数セットの対応関係の一例を示す図である。

【図33】ソフトウェア的にフィルタ演算を行う装置の一構成例を示すブロック図である。

【図34】図33の装置において、画像の拡大を行う場合における入力データの供給のパターンの一例を示す図である。

【図35】処理に必要なデータを有する要素プロセッサとの位置関係の例を示す図である。

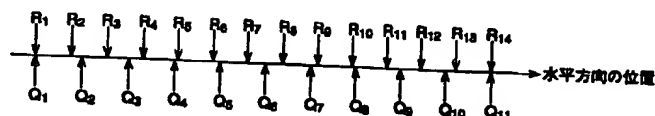
【符号の説明】

5 制御回路, 1.1 剰余回路, 1.2 レジスタ,
1.3 近似回路, 2.1 入力ポインタ, 2.2 入力SAM部,
2.3 データメモリ部, 2.4 ALUアレイ部, 2.5 出力SAM部, 2.6 出力ポインタ,
2.7, 2.7A, 2.7B, 2.7C, 2.7D プログラム制御部, 2.8, 2.8A, 2.9 メモリ

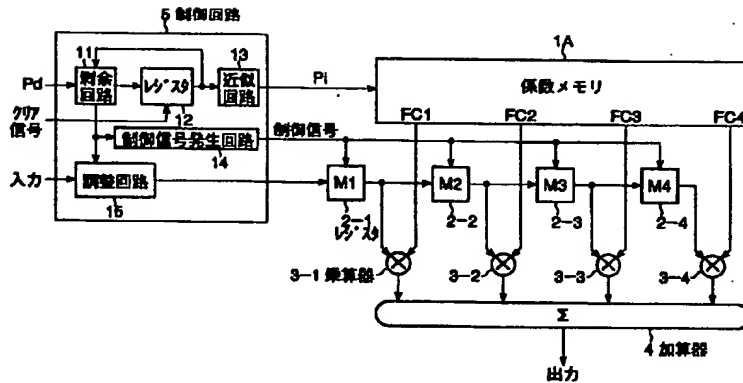
【図10】

画素	左2画	左1画	自分	右1画	右2画	右3画
2		○	○		○	
4	○		○	○	○	○

【図26】



【図1】



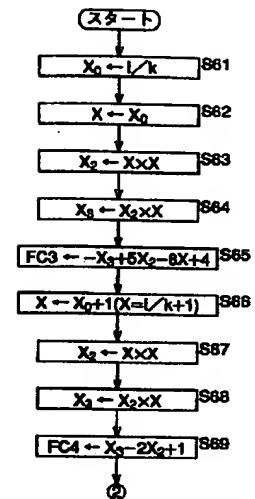
【図2】

フィルタ 選択信号P	正帰化 位相遅延x	1/秒係数k(小数点部係数)				1/秒係数k(整数部係数)			
		Fc1	Fc2	Fc3	Fc4	Fc1	Fc2	Fc3	Fc4
P ₀	0	0.0	1.0	0.0	0.0	0	128	0	0
P ₁	1/16	-0.0625	-0.0649	0.9824	0.0682	-0.0037	-7	127	8
P ₂	2/16	-0.125	-0.0857	0.8707	0.1387	-0.0137	-12	124	18
P ₃	3/16	-0.1875	-0.1238	0.6363	0.2161	-0.0288	-18	120	28
P ₄	4/16	-0.25	-0.1408	0.3906	0.2989	-0.0469	-18	114	38
P ₅	5/16	-0.3125	-0.1477	0.1352	0.3796	-0.0671	-19	107	49
P ₆	6/16	-0.375	-0.1485	0.7715	0.4629	-0.0879	-19	99	59
P ₇	7/16	-0.4375	-0.1384	0.7009	0.5452	-0.1077	-18	90	70
P ₈	8/16	-0.5	-0.1250	0.6250	0.6250	-0.1250	-16	80	80
P ₉	9/16	-0.5625	-0.1077	0.5452	0.7009	-0.1384	-14	70	90
P ₁₀	10/16	-0.625	-0.0879	0.4629	0.7715	-0.1485	-11	59	99
P ₁₁	11/16	-0.6875	-0.0671	0.3796	0.8352	-0.1477	-9	49	107
P ₁₂	12/16	-0.75	-0.0469	0.2989	0.8707	-0.1387	-6	38	114
P ₁₃	13/16	-0.8125	-0.0288	0.2161	0.9363	-0.1238	-4	28	120
P ₁₄	14/16	-0.875	-0.0137	0.1387	0.9707	-0.0857	-2	18	124
P ₁₅	15/16	-0.9375	-0.0037	0.0682	0.9824	-0.0549	0	8	127

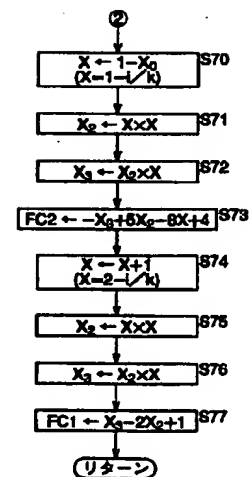
【図9】

種類	左2周	左1周	自分	右1周	右2周	右3周	例
1	○	→	○	○	→	○	Q ₃ , Q ₆ , Q ₁₃ , ...
2		○	○		○	○	Q ₄ , Q ₇ , Q ₁₁ , ...
3	○	○	→	○	○		Q ₅ , Q ₈ , Q ₁₂ , ...
4	○		○	○	○		Q ₉ , ...
5		○	○	○	○	○	Q ₁₀ , ...

【図16】



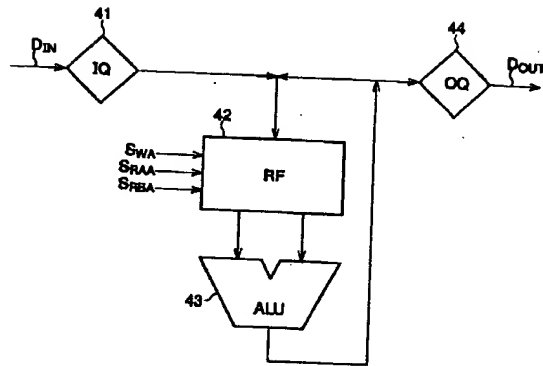
【図17】



【図3】

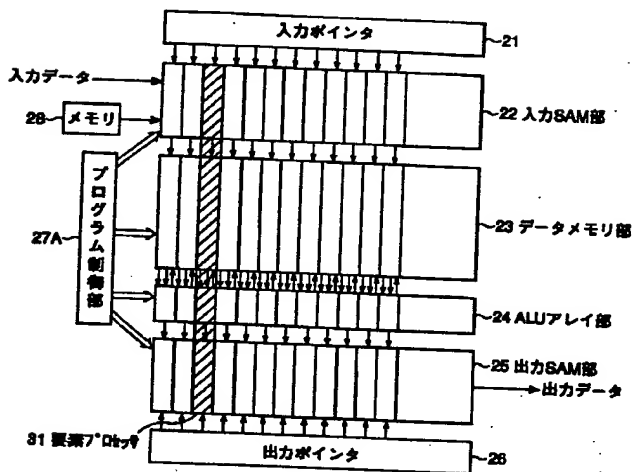
Qi	IPd	剰余 (位相)	PI (D-16)	位相誤差
Q1	0.0	0.0	$P_0 (x=0.0)$	0.0
Q2	0.7	0.7	$P_{11} (x=0.6875)$	0.0125
Q3	1.4	0.4	$P_8 (x=0.375)$	0.025
Q4	2.1	0.1	$P_2 (x=0.125)$	0.025
Q5	2.8	0.8	$P_{13} (x=0.8125)$	0.0125
Q6	3.5	0.5	$P_9 (x=0.5)$	0.0
Q7	4.2	0.2	$P_3 (x=0.1875)$	0.0125
Q8	4.9	0.9	$P_{14} (x=0.875)$	0.025
Q9	5.6	0.6	$P_{10} (x=0.625)$	0.025
Q10	6.3	0.3	$P_6 (x=0.3125)$	0.0125
Q11	7.0	0.0	P_0	
Q12	7.7	0.7	P_{11}	
Q13	8.4	0.4	P_8	
Q14	9.1	0.1	P_2	
Q15	9.8	0.8	P_{13}	
Q16	10.5	0.5	P_9	
...				

【図5】

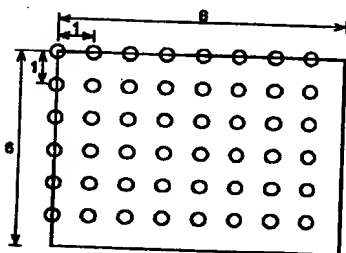


【図7】

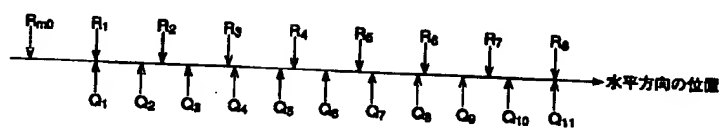
【図4】



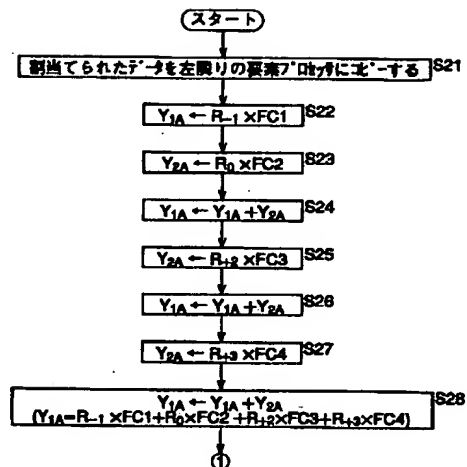
【図21】



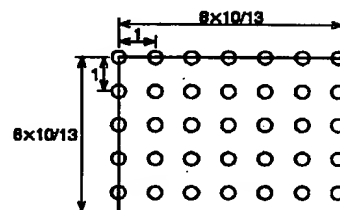
【図23】



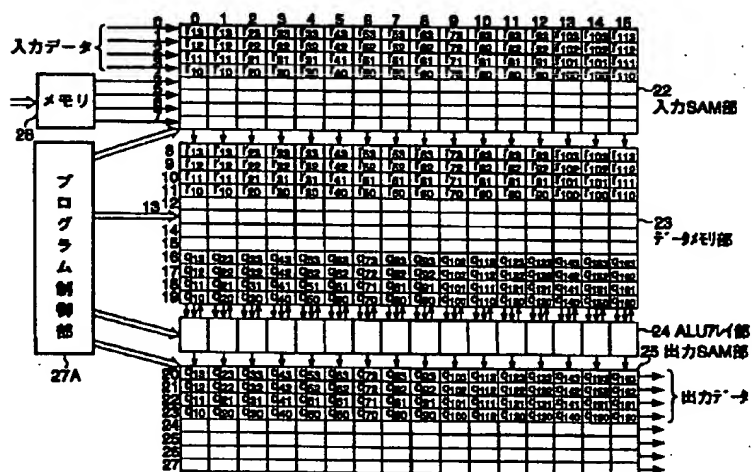
【图 1 1】



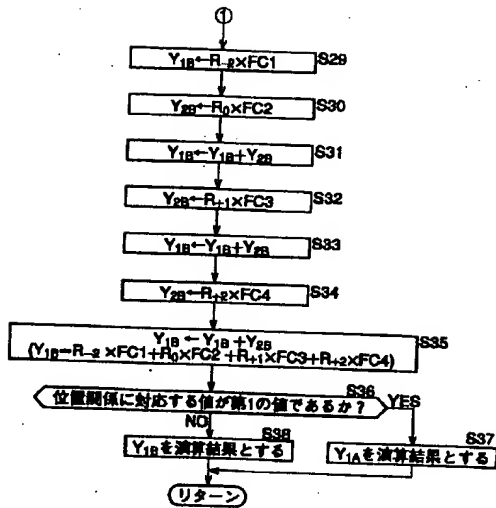
【图 2 5】



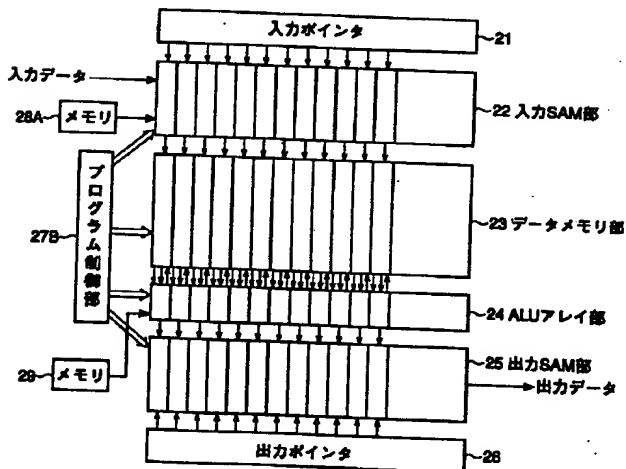
【图8】



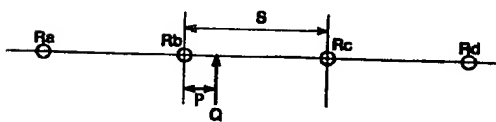
【図12】



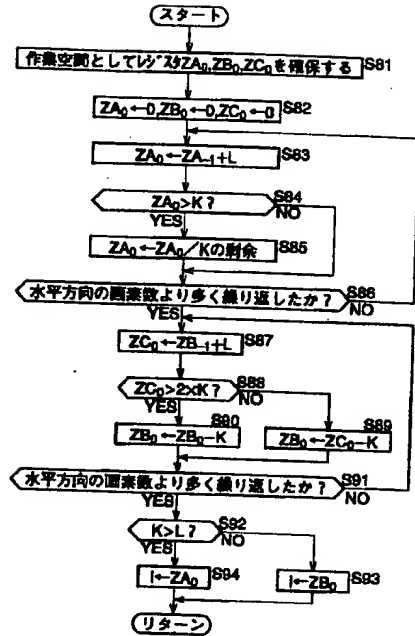
【図13】



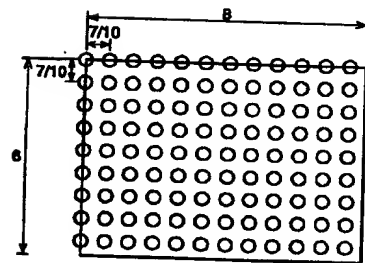
【図28】



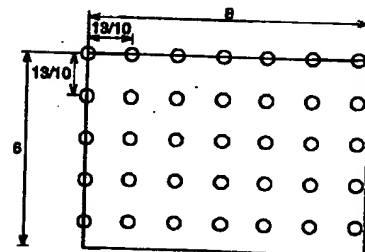
【図19】



【図24】



【図27】

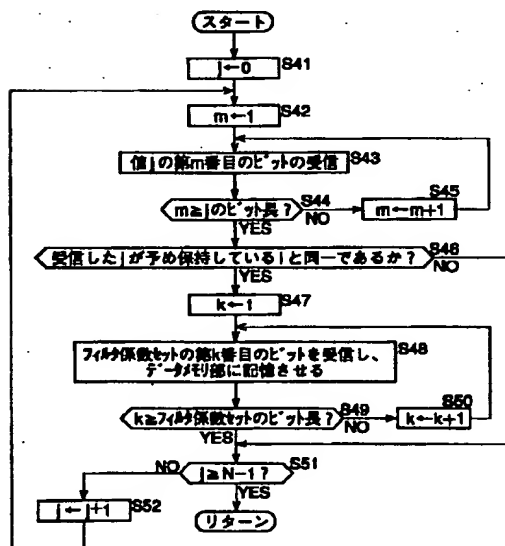


【図14】

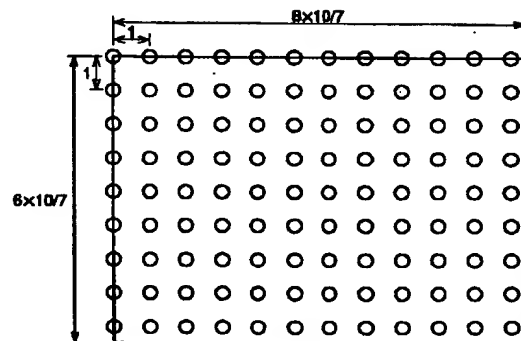
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
8																
9																
10																
11																
12	ϕ_{03}	ϕ_{113}	ϕ_{03}	ϕ_{23}	ϕ_{133}	ϕ_{03}	ϕ_{33}	ϕ_{143}	ϕ_{103}	ϕ_{53}	ϕ_{03}	ϕ_{113}	ϕ_{03}	ϕ_{23}	ϕ_{133}	ϕ_{03}
13	ϕ_{02}	ϕ_{112}	ϕ_{02}	ϕ_{22}	ϕ_{132}	ϕ_{02}	ϕ_{32}	ϕ_{142}	ϕ_{102}	ϕ_{52}	ϕ_{02}	ϕ_{112}	ϕ_{02}	ϕ_{22}	ϕ_{132}	ϕ_{02}
14	ϕ_{01}	ϕ_{111}	ϕ_{01}	ϕ_{21}	ϕ_{131}	ϕ_{01}	ϕ_{31}	ϕ_{141}	ϕ_{101}	ϕ_{51}	ϕ_{01}	ϕ_{111}	ϕ_{01}	ϕ_{21}	ϕ_{131}	ϕ_{01}
15	ϕ_{00}	ϕ_{110}	ϕ_{00}	ϕ_{20}	ϕ_{130}	ϕ_{00}	ϕ_{30}	ϕ_{140}	ϕ_{100}	ϕ_{50}	ϕ_{00}	ϕ_{110}	ϕ_{00}	ϕ_{20}	ϕ_{130}	ϕ_{00}
16																
17																
18																
19																

23
ビットメモリ部

【図15】



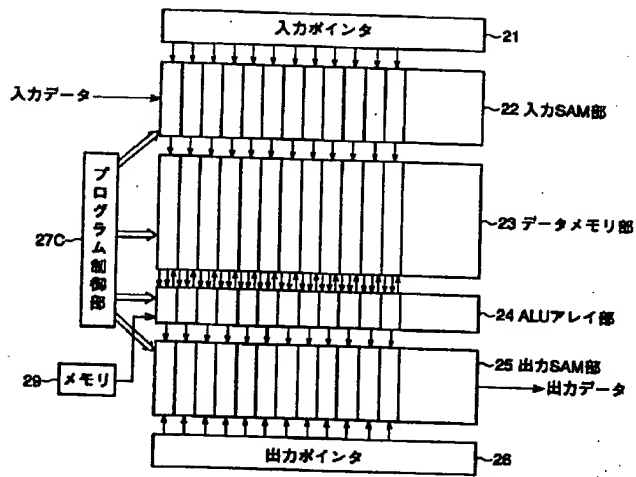
【図22】



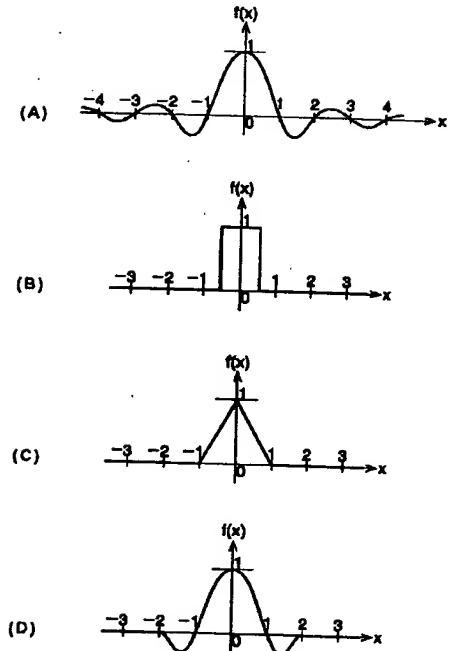
【図35】

	種類	左2隣	左1隣	真上	右1隣	右2隣	右3隣	例
1		○		○	○		○	Q_3, Q_6, Q_{13}, \dots
2			○	○		○	○	Q_4, Q_7, Q_{11}, \dots
3		○	○		○	○		Q_5, Q_8, Q_{12}, \dots
4		○		○	○	○		Q_9, \dots
5			○	○	○		○	Q_{10}, \dots

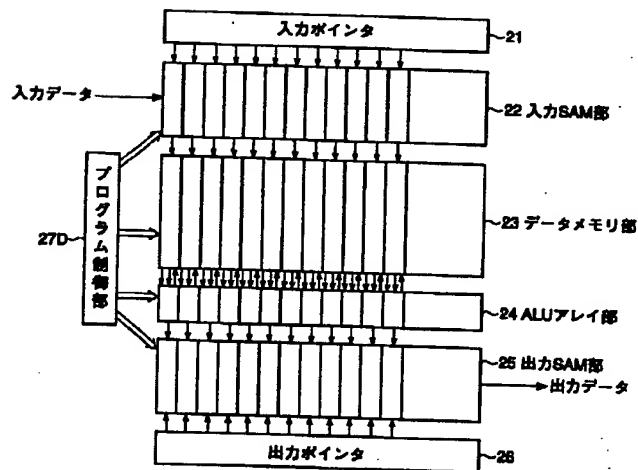
【図18】



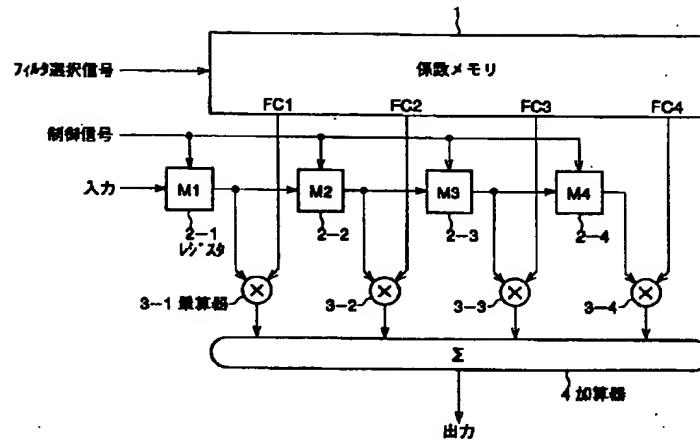
【図29】



【図20】



【図30】



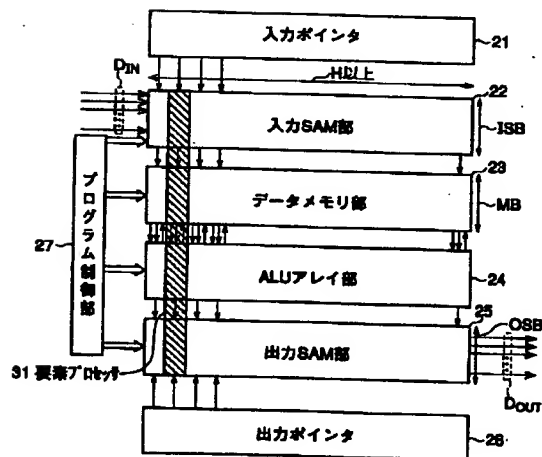
【図31】

サイクル	入力データ	ファイル選択信号	制御信号	レジスタ2-1乃至2-4				出力データ
1	R ₁		H	R _{m0}	R _{m1}	R _{m2}	R _{m3}	
2	R ₂		H	R ₁	R _{m0}	R _{m1}	R _{m2}	
3	R ₃		H	R ₂	R ₁	R _{m0}	R _{m1}	
4	R ₄	P ₀	L	R ₃	R ₂	R ₁	R _{m0}	Q ₁
5	R ₄	P ₇	H	R ₃	R ₂	R ₁	R _{m0}	Q ₂
6	R ₅	P ₄	H	R ₄	R ₃	R ₂	R ₁	Q ₃
7	R ₆	P ₁	L	R ₅	R ₄	R ₃	R ₂	Q ₄
8	R ₆	P ₆	H	R ₅	R ₄	R ₃	R ₂	Q ₅
9	R ₇	P ₅	H	R ₆	R ₅	R ₄	R ₃	Q ₆
10	R ₈	P ₂	L	R ₇	R ₆	R ₅	R ₄	Q ₇
11	R ₈	P ₉	H	R ₇	R ₆	R ₅	R ₄	Q ₈
12	R ₉	P ₆	H	R ₈	R ₇	R ₆	R ₅	Q ₉
13	R ₁₀	P ₃	L	R ₈	R ₇	R ₆	R ₅	Q ₁₀
14	R ₁₀	P ₀	H	R ₉	R ₈	R ₇	R ₆	Q ₁₁

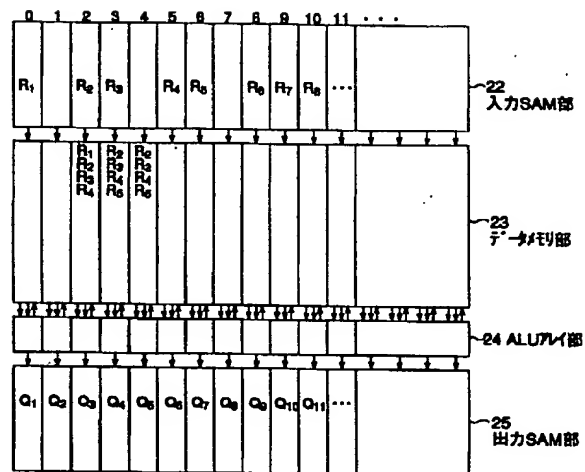
【図32】

フィルタ 選択信号 P_i	正規化 位相量 x	小数点表現係数				8ビット表現係数				係数和
		FC1	FC2	FC3	FC4	FC1	FC2	FC3	FC4	
P_0	$\frac{0}{10} = 0.0$	0.0	1.0	0.0	0.0	0	128	0	0	128
P_1	$\frac{1}{10} = 0.1$	-0.009	0.981	0.109	-0.081	-1	126	14	-10	126
P_2	$\frac{2}{10} = 0.2$	-0.032	0.928	0.232	-0.128	-4	119	30	-16	126
P_3	$\frac{3}{10} = 0.3$	-0.063	0.847	0.363	-0.147	-8	106	46	-18	127
P_4	$\frac{4}{10} = 0.4$	-0.096	0.744	0.496	-0.144	-12	85	63	-18	128
P_5	$\frac{5}{10} = 0.5$	-0.125	0.625	0.625	-0.125	-16	60	80	-16	128
P_6	$\frac{6}{10} = 0.6$	-0.144	0.496	0.744	-0.096	-16	63	95	-12	126
P_7	$\frac{7}{10} = 0.7$	-0.147	0.363	0.847	-0.063	-19	46	106	-8	127
P_8	$\frac{8}{10} = 0.8$	-0.128	0.232	0.928	-0.032	-16	30	119	-4	126
P_9	$\frac{9}{10} = 0.9$	-0.081	0.109	0.981	-0.009	-10	14	126	-1	126

【図33】



【図34】



フロントページの続き

(72)発明者 中村 憲一郎
 東京都品川区北品川6丁目7番35号 ソニ
 ー株式会社内

THIS PAGE BLANK (USPTO)